Old document recognition using fuzzy methods

J.M.C. Sousa*, J.M. Gil, C.S. Ribeiro and J.R. Caldas Pinto

Department of Mechanical Engineering, Instituto Superior Técnico,
Technical University of Lisbon,
GCAR/IDMEC 1049-001 Lisbon, Portugal
E-mail: jmsousa@ist.utl.pt E-mail: jgil@ext.bn.pt
E-mail: crib@ext.bn.pt E-mail: jcpinto@dem.ist.utl.pt
*Corresponding author

Abstract: This paper proposes an expert system based on fuzzy logic for optical character recognition of old printed documents. These documents can have some problems, such as distortion, poor printing quality, faded and misprinted characters, speckles and smudges. The recognition process consists of two stages: training with character image examples and classification of new character images. The proposed OCR builds fuzzy membership functions from oriented features extracted using Gabor filter banks. The proposed methodology is tested on three different books from the 17th century, written in Portuguese. The fuzzy recogniser presents a very high character recognition success rate, which confirms the advantage of using expert systems in image based decision systems.

Keywords: fuzzy OCR; character recognition; old documents.

Reference to this paper should be made as follows: Sousa, J.M.C., Gil, J.M., Ribeiro, C.S. and Pinto, J.R.C. (2006) 'Old document recognition using fuzzy methods', *Int. J. Intelligent Systems Technologies and Applications*, Vol. 1, Nos. 3/4, pp.263–279.

Biographical notes: J.M.C. Sousa was born in 1966 in Lisbon, Portugal. He received his MSc degree in Mechanical Engineering from the Technical University of Lisbon, in 1992, and received his PhD in Electrical Engineering from the Delft University of Technology, Netherlands in 1998. He is currently an Assistant Professor at the Control, Automation and Robotics Group, Department of Mechanical Engineering at Instituto Superior Técnico, Technical University of Lisbon. His main research interests include fuzzy model-based control, intelligent control, optimisation and fuzzy decision making. He is leading the Intelligent Automation Group, Centre of Intelligent Systems, Institute of Mechanical Engineering, Portugal.

João M. Gil was born in 1980 in Lisbon, Portugal and graduated from Instituto Superior Técnico in Computer Engineering in 2003. He developed his final course project about optical character recognition on ancient documents associated with the Innovation and Development Services of the National Library of Portugal. He currently works for this same Department at the National Library of Portugal and as an investigator at INESC-ID, working on image processing, indexing, character recognition and digital work structuring. He is also interested in digital image rendering and video and sound processing.

Cláudia S. Ribeiro was born in 1980 in Lisbon, Portugal and graduated from Instituto Superior Técnico in Computer Engineering in 2003. She developed her final course project about optical character recognition on ancient documents associated with the Innovation and Development Services of the National Library of Portugal. After graduation she worked for this same Department at the National Library of Portugal developing software to simplify several manual tasks related to image processing, organisation and web access. She currently works as a consultant in the telecommunications field.

J.R. Caldas Pinto was born in Leiria, Portugal in 1951 and graduated from Instituto Superior Técnico, Lisbon in 1974. He received his PhD in control systems in Manchester in 1983. He is Associate Professor at the Instituto Superior Técnico. His research interests include image processing and pattern recognition, principally with respect to old documents, and vision based control chiefly as it applies to robotics.

1 Introduction

Optical Character Recognition (OCR) is a practical application of state-of-the-art image processing and pattern recognition developments (Mori et al., 1999). Uses of OCR include digital document archiving, printed text search and automated form processing. Current communication facilities could allow broad and public distribution of vast libraries of books, newspapers, magazines and all kinds of printed media, if quality, cost-effective OCR procedures are available for mass digitising.

While modern printed text can be recognised very accurately with commercially available software, performing OCR on more exotic material (such as gothic fonts, ancient typesets and handwriting) is currently and noticeably less successful (Jain and Lazzerini, 1999).

This paper explores the combination of expert systems and image-based character recognition of old printed documents, proposing a fuzzy recogniser specifically tailored to this type of documents and corresponding typesets. The proposed algorithm is a development of a handwriting word recognition system using fuzzy logic (Buse et al., 2001). The use of fuzzy classification (Sousa and Kaymak, 2002) improves results by providing larger tolerance for unstable typesetting and printing technologies.

The recogniser is based on an analytic perspective, i.e., it considers each character separately. Building a holistic recogniser able to handle nearly any text in full would require training with virtually every single word in a language, demanding enormous memory resources and taking an unacceptably long time to classify each word. Holistic recognition is far better suited for mass indexing by a few known, relevant words; generic OCR using such a system is unrealistic with current technology. Ancient documents are not so difficult as handwritten recognition, but clearly more difficult than standard font OCR. OCR is certainly a very useful tool to manipulate information of old documents in a digital format. However, OCR of ancient documents should take into account their specificities.

The recognition algorithm proposed in this paper is especially suited for old documents, and it works in two steps. The first step, training, considers sets of character images, known as character groups, and combines their dominant graphical features, resorting to Gabor filter banks to execute oriented feature extraction (Buse et al., 1997).

These composite images are then used to build fuzzy membership functions that, in a sense, describe the visual attributes of every character group (Buse et al., 2001). The second step, classification, is where the actual recognition takes place. A new character image is processed by Gabor filters and normalised, and then, it is compared to the training results. The closest match, dictated by a fuzzy decision maker, is returned as the most likely classification.

This paper is divided as follows. First, the problem of character recognition of old documents is discussed in Section 2. Then, the fuzzy recogniser is described in Section 3. This section presents the global recognition system overview. Further, both the training and the classification are described. The results for the proposed recogniser are presented in Section 4, where it is shown that the algorithms proposed in this paper are superior to one of the best commercial OCR packages, as can be seen in ABBYY OCR (2005). Finally, some conclusions are drawn in Section 5.

2 Character recognition of old documents

Commercial OCR packages are widely available today and work exceedingly well on quality scans of modern typeset text. Most optical and geometric corrections needed for successful recognition are handled and performed automatically. Format retention, which is the ability to identify and recreate formatting and layout information, and recognition of more elaborate structures, such as tables and captions, is now possible with some leading systems. Recognition accuracy for these cases is so high that most commercial OCR software development is aimed at improving speed, user interface and advanced features, instead of dealing with more 'complicated' situations.

The old documents that this paper deals with, dating as far back as the 17th Century, are an example of such more 'complicated' situations. This is due to a number of issues and problems that arise at various stages of the recognition process. Firstly, the scanning itself is usually not perfect; ancient books must be handled with care and some amount of distortion is unavoidable. The geometric and lighting irregularities that ensue should be corrected in a preprocessing stage. Secondly, paper and printing quality is often poor. Faded and misprinted characters, irregular character and word spacing, speckles and smudges are quite common, as well as seeing through to back pages. These problems are illustrated in Figure 1 and can be partially addressed at a preprocessing level, but should also be tolerated by flexible and powerful recognition algorithms.

Figure 1 Paper and printing problems in old documents (a) faded character; (b) back page see through and (c) speckle

(a) (b) (c)

Ancient spelling can also be significantly different from modern spelling, thus making it harder for standard spell checkers and correctors to perform suitably. Finally, the ancient typesets themselves include characters different from those in modern print and can have

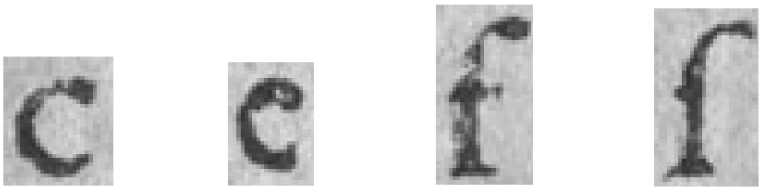
a significantly distinct visual aspect. The simple use of commercial OCR software on such a difficult context yields results that range from good to virtually unreadable, clearly stating that there is room for improvement and further development.

Geometric and optical issues, such as skewing, text orientation and tilt, as well as cleaning specks and smudges, are not the target of this paper. Instead, focus is on finding ways of recognising text, compensating for the quality limitations of the source material and adjusting and expanding current OCR systems in order to handle old documents specifically. Possibly, the most significant problems faced concern character and word spacing and differentiating between certain key characters.

Spacing in ancient documents is very irregular, both between characters and between words. Erratic character spacing, associated with faded or faulty print, lead to the splitting of certain characters into two or to the joining of separate characters (for instance, 'm' and 'rn').

Besides all other printing flaws and typeset particularities, there are a few characters that can be especially mistaken. Figure 2 shows two of these, namely the 'f', elongated 's', 'e' and 'c'. The elongated 's' is no longer used in modern print and is very similar to the standard 'f', as much so that even human readers have trouble telling the two characters apart. The 'e' and 'c' characters are mistaken mostly because of faulty print, rendering the horizontal dash in the 'e' not clearly visible or the upper part of the 'c' too thick. These are just two examples, but they alone accounted for over 25% of recognition errors in early results. This paper proposes the use of a fuzzy recogniser to attenuate these problems, as proposed in the following.

Figure 2 Easily mistaken characters: 'c', 'e', 'f' and 's'



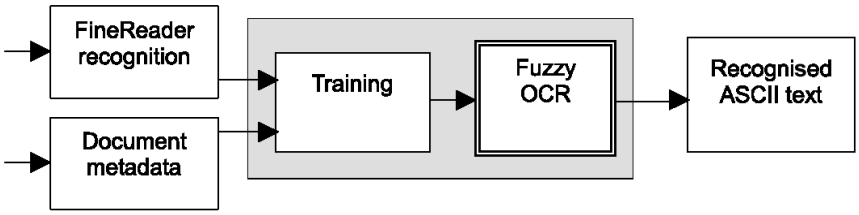
3 Fuzzy recogniser

3.1 System overview

This section gives an introduction to the general functioning of the developed application. Figure 3 displays a diagram representing schematically the organisation of the developed recognition system and the streamlined design connecting its components. The segmentation to obtain each individual character image is performed using the OCR package ABBYY FineReader Engine (ABBYY Software House, 2006). This package also provides methods to segment entire words, according to the computed geometric information, as well as its own OCR output, which was used as a base of comparison with the fuzzy recognition system. Note that this is one of the most advanced commercial packages for OCR. Moreover, we are using a development license containing the most recent advances in the software. Therefore, our implementation is compared to the most recent state-of-the-art OCR techniques. The information obtained from the FineReader

OCR is used to build a manually classified character database, which is applied in the training stage to build models for every known character. When a new image is given for recognition, the most likely classification is given to each segmented character, thus performing the intended OCR on the characters.

Figure 3 Diagram of the fuzzy document recogniser



The proposed OCR recognises characters instead of words, which was proposed in Buse et al. (2001). This change reduces resource requirements; character images are smaller, so processing takes less time and the size of the training data structure is easily handled by modern hardware. Further, recogniser parameters and thresholds are configured differently as well, adjusted to provide a finer and more comprehensive analysis of character features. Other original changes include the disabling of the alignment process and an additional aspect ratio classification factor, which are explained in the following.

The recognition process requires a previous training step, followed by the intended character classification. These two stages are presented in this section.

3.2 Training

The training process is performed in two steps: oriented feature extraction, where the dominant features of a character are extracted using Gabor filter banks, and membership function generation, which are generated for each character group and for each orientation, based on the extracted features of the training images.

The dominant features of a character consist of what is more common not-to-change between the typing styles, as the long vertical stroke in the b's and t's, for example. In this paper, their extraction is performed through Gabor filter banks, which allow oriented feature extraction. Each filter, oriented at a given angle ϕ, extract the features of a character.

3.2.1 Feature extraction

The oriented feature extraction is performed using Gabor filters. The Gabor (1946) filter is a typical wavelet that offers localised operations. The result of the filtering can be used to extract local information from regions of the image, in time and frequency domains, and it can achieve minimum uncertainty in both of them (Buse et al., 1997). The Gabor filter is defined in a spacial (x, y) and in a frequency (u,v) domains as, respectively:

$$g(x,\ y) = \exp\left\{-\pi\left(\frac{x'^2}{\sigma_x^2} + \frac{y'^2}{\sigma_y^2}\right)\right\} \times \exp\{2\pi j(u_0 x + v_0 y)\}$$
$$G(u,v) = \exp\{-\pi[(u' - u_0')^2\sigma_x^2 + (v' - v_0')^2\sigma_y^2]\} \tag{1}$$

where

$$x' = x\cos\phi + y\sin\phi \quad y' = -x\ \sin\phi + y\cos\phi$$
$$u' = u\cos\phi + v\sin\phi \quad v' = -u\ \sin\phi + v\cos\phi$$
$$u_0' = u_0\cos\phi + v_0\sin\phi \quad v_0' = -u_0\sin\phi + v_0\cos\phi$$
$$u_0 = f\cos\theta \quad v_0 = f\sin\theta$$
$$j = \sqrt{-1} \quad f = \sqrt{u_0^2 + v_0^2}.$$

Equation (1) represents a 2D Gaussian centred at (u_0,v_0) in the frequency domain. The parameters σ_x and σ_y are the standard deviations of the 2D Gaussian, determining the frequency and orientation bandwidths of the filter. The angle ϕ defines the Gabor wavelet direction, and the angle $\theta = \phi + 90°$ defines the wavelet orientation. The frequency f determines the distance to the origin of the image frequency spectrum.

The spacing between the Gabor angles $\Delta\phi$ is an important parameter. However, its value is not critical in optimising the shape of the Gabor filters for extracting parts. These parameters are character dependent, which means that the estimation has to take into account, e.g., the size of the characters and the thickness of the writing (Buse et al., 1997). For this reason, usually, their values are selected on a trial-and-error basis. For this paper, 12 wavelet directions were considered, from 0° to 180°, as in Buse et al. (2001).

Features are extracted by first applying a discrete Fourier transform to the image. The resulting output is processed to avoid spurious features. Because this output is complex, the power image is used; the parts oriented at the direction ϕ_i have a larger intensity than the parts oriented away from that direction. The image is normalised to produce consistent results among distinct cases, by resizing each image to the largest dimension among each character group.

The word recognition in Buse et al. (2001) required a time-consuming alignment algorithm in order to match extracted word features among word group samples. This procedure is needed, especially for handwritten text, to compensate for variations in character spacing and shape, and to normalise word bounds. This procedure is not implemented in this paper. Notice that characters instead of words are being recognised. Character alignment can produce heavy distortion when feature match is not effective. A single character has a smaller number of dominant features; printing flaws, common in this context, complicate feature-matching even further. Character segmentation is also tight and accurate from the beginning. This simplification reduces the computational effort significantly.

As each training sample of the same character contains essentially the same extracted features structure, the major structural components can be established by adding the standardised images together, which form the composite image for each trained character. Character images are classified manually; ancient characters are labelled as their present-day equivalent, therefore solving the problem of generating standard

ASCII text from these occurrences. An example of original images is given in Figure 4. Figure 5 presents the image resulting from applying Gabor filters with 90° orientation.

Figure 4 Original character images

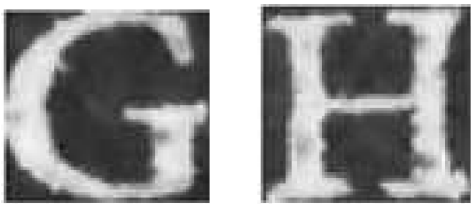
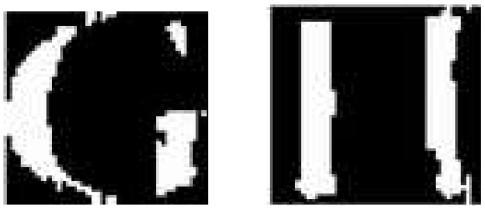


Figure 5 Feature extraction using Gabor filters



Besides the information in the Gabor filters, in this paper, information regarding image aspect ratios for the characters is also stored. The aspect ratio of a given image I is defined as

$$ar(I) = \frac{w(I)}{h(I)} \tag{2}$$

where $w(I)$ is the image width and $h(I)$ is the image height. The average aspect ratio for each character group j is defined as:

$$ar_j = \frac{\sum_{k=1}^{N_j} ar(I_{jk})}{N_j} \tag{3}$$

where I_{jk} is the kth sample image for character group j and N_j is the total number of images for group j. Aspect ratio values ar_j are used in classification as further assistance in the identification of unknown characters.

3.2.2 Membership function generation

Before the classification process can take place, a set of fuzzy membership functions is generated for each character group and for each orientation, based on the extracted features of the training images. The membership functions intend to provide a description of image features for use within the recognition algorithm.

Each training sample of a character group C_j contains the main features structure. Thus, a composite image R_{ij}, can be constructed for each point (pixel) (x,y) as follows:

$$R_{ij}(x,y) = \frac{1}{N_j}\sum_{k=1}^{N_j} C_{ijk}(x,y), \tag{4}$$

where i indicates an orientation (as e.g., the ones of the Gabor filters) and k is a sample image of the jth character group.

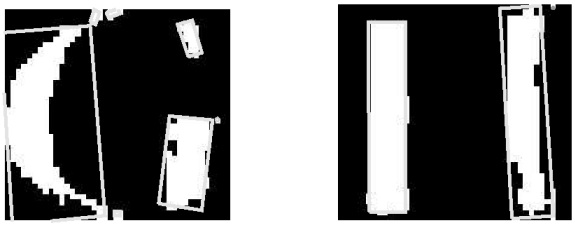
The membership functions are represented by twisted trapezoids. This shape is defined by two oriented bounded rectangles. This shape provides a better fit to the shape of the data than 2-D trapezoids (Buse et al., 2001). Upper and lower thresholds must be defined for the upper and lower rectangles, based on the composite images R_{ij}. The upper, H_u, and the lower, H_l, thresholds are used in the binarisation of the image (Parker, 1998), thus finding the upper and lower boundaries for each feature. These thresholds are determined by:

$$H_u = c_u \max\{R_{ij}(x,y)\} \tag{5}$$

$$H_l = c_l \max\{R_{ij}(x,y)\}, \tag{6}$$

where $R_{ij}(x,y)$ was defined in equation (4), and constants c_u and c_l are set at 0.4 and 0.25, respectively. These values were determined empirically in Buse et al. (2001) and are quite successful, but they can also be based on a standard binarisation method (Otsu, 1979) for more demanding cases and greater robustness. The two bounding rectangles are determined around the extracted features, as proposed in Toussaint (1983), corresponding to each intensity threshold. The bounding rectangles resulting from the upper thresholds defined in equation (5) for the features extracted using Gabor filters, presented in Figure 5, are shown in Figure 6.

Figure 6 Bounding rectangles around extracted features



Each rectangle pair is used to build a partial membership function. Its value $\mu(x,y)$ is 1 in the inner rectangle area and zero outside the outer rectangle. The vertices of a rectangle pair are linked based on minimisation of the Euclidean distance. The intermediate function values are interpolated, forming a twisted trapezoidal shape (Foley et al., 1990; Buse et al., 2001). To do so, the function domain is divided into 13 regions, as represented in Figure 7. Region 1 in this figure has the following membership value:

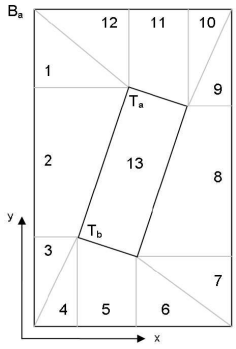
$$\mu(x,y) = \frac{x - x_{B_a}}{x_{T_a} - x_{B_a}} \tag{7}$$

where x and y are the coordinates of a given point, B_a, T_a and T_b are the points shown in Figure 7, which are described by their respective coordinates x and y. Similar expressions can be found for regions 3, 4, 6, 7, 9, 10 and 12. The membership function for region 2 in Figure 7 has the following values:

$$\mu(x,y) = \frac{(x - x_{B_a})(y_{T_b} - x_{T_a})}{(y - x_{T_a})(x_{T_b} - x_{T_a})(x_{T_a} - x_{B_a})(y_{T_b} - y_{T_a})}. \tag{8}$$

Similar interpolation rules are applied to regions 5, 8 and 11. Thus, the expressions for regions 3–12 are analogous and can be obtained by considering the rectangle vertices in sequence and swapping the coordinate pairs. In region 13, the membership value is naturally $\mu(x,y) = 1$.

Figure 7 Domain regions of the membership functions

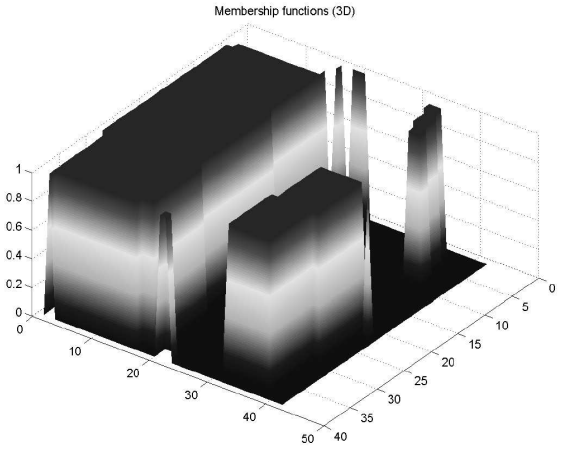


The global membership function for a given orientation i, denoted as $a_{ij}(x, y)$, is defined as the maximum of the partial membership functions at each point:

$$a_{ij}(x,y) = \max(\mu_{ijl}(x,y)), \tag{9}$$

where j denotes a character group and l is a membership function. Thus, in cases in which features overlap equation (9), it takes only into account the most relevant feature. These maximum values are found continuously during the membership function generation process, in order to minimise resource usage. The membership function generated for the letter 'G' and the orientation of the Gabor filter equal to 90° in Figure 4 is presented in Figure 8.

Figure 8 Membership functions for example in Figure 4



3.3 Classification

Once the training phase is complete, it is possible to classify new unseen characters. The input for this stage is a character image C and the global membership functions built during the training phase described in Section 3.2. First, the image is preprocessed in order to extract its features. Namely, it is filtered through Gabor filter banks and normalised, exactly as applied in the training phase. The purpose of this step is to match the features of the processed image to those of the training images, resulting in a set of feature images C_i. This set is compared, using a fuzzy decision process, to the training character groups and their respective membership functions.

A similarity rating can be computed between the test character and the membership functions of the training characters. A larger similarity should translate a bigger match between the character image being evaluated C and the training character images C_j. The largest similarity indicates the closest match, and the input character is classified as belonging to the character group with higher similarity value. This similarity S_{ij} is defined as:

$$S_{ij}(C) = S(C, \phi_i, C_j) \tag{10}$$

i.e., S_{ij} is a function of the input character C to be classified, the angles of the Gabor filters ϕ_i and the training character groups C_j. The details of this calculation are defined in the following.

The similarity measure is calculated using a weighted average of the global membership functions $a_{ij}(x,y)$ defined in equation (9), and the intensity value of an input character C_i for orientation i, normalised to one, which is denoted as C'_{ij}. Considering that the weight of a pixel (x,y) is denoted as $w_{ij}(x, y)$, the similarity is given by:

$$S_{ij}(C) = \frac{\sum_{x,y} w_{ij}(x,y) a_{ij}(x,y) C'_{ij}(x,y)}{\sum_{x,y} w_{ij}(x,y)}. \tag{11}$$

The weights $w_{ij}(x,y)$ are assigned to each image point (x,y) for each orientation i and each character group j, to measure its influence, related mostly to the membership function values. The weights are calculated according to:

$$w_{ij}(x,y) = \begin{cases} C'_{ij}(x,y) & \text{if } a_{ij} = 0 \\ w'_{ij}(x,y) & \text{if } a_{ij} \neq 0 \end{cases} \tag{12}$$

where $C'_{ij}(x,y)$ is a point in the normalised test character (location of the point (x,y) in the test character C'_{ij}).

The upper term in equation (12) assures that points, where $C'_{ij}(x,y)$ is not zero but the membership function is, will be penalised. This happens when a feature in $C'_{ij}(x,y)$ does not match any orientation i for character group j. In this case, the value increases the denominator of equation (11), while not affecting the numerator, and therefore lowers the computed similarity rating.

The lower term in equation (12) is the rate of significance of point (x,y), and is denoted by $w'_{ij}(x,y)$. Considering that N_c is the total number of character groups, the weighting function is given by:

$$w'_{ij}(x,y) = \begin{cases} 0, & \text{if } a_{ij} = 0 \\ \dfrac{N_c N_+}{N_+(N_c - 1)} \sum_{j=1}^{N_c} a_{ij}(x,y) & \text{if } a_{ij} \neq 0 \end{cases} \tag{13}$$

where N_+ is the number of character groups, for each orientation and each point, with a positive membership grade $a_{ij}(x,y)$. It attempts to formalise the intuitive concept that point (x,y) is a distinguishing factor among character groups, when only a restricted set of groups has membership values for that point.

The similarity ratings proposed in equation (11) are determined considering the need to penalise the value of points with high $w_{ij}(x,y)$ but low or zero $a_{ij}(x,y)$ or C'_{ij}. In these cases, the image point has non-zero intensity outside the membership function area or low or zero intensity within the membership function area. It should be noted that the similarity measure in equation (11) is similar to the one proposed in Buse et al. (2001). However, that paper used a somewhat different and more complex formal notation, in part to account for the formal distinction between the various partial membership functions for each word and orientation. In this paper, the partial functions, each corresponding to a particular feature, were previously combined, which leads to a simpler notation and the improvement of computational resource usage.

Several methods can be used to aggregate the similarity values, S_{iy} (Chen et al., 1992), from which the simple additive weighted method (Hwang and Yoon, 1981) is one of the most utilised; it was used in the holistic recogniser presented in Buse et al. (2001). In this paper, we propose a different aggregation method, which leads to better classification results. The decision is based on the normalised values for the global membership functions for a given orientation *i*, and a character group *j,* which is defined as v_{ij}:

$$u_{ij} = \frac{\sum_{x,y} a_{ij}(x,y)}{\sum_{i,x,y} a_{ij}(x,y)}. \tag{14}$$

This value is the ratio between membership function volume for a given orientation *i*, and the total function volume for every orientation, within a character group. The rationale is that orientations with a relatively larger membership function volume should have a greater influence in the decision-making.

The similarity values are normalised, and denoted S'_{iy}, with respect to the orientations *i*:

$$S'_{ij} = \frac{S_{ij}}{\max_i(S_{ij})}. \tag{15}$$

This paper introduces a new *aspect factor* $r_j(C)$, which compares the aspect ratio of the character image *C* with the average aspect ratio ar_j of character group *j*. Recall that the aspect ratio was defined in equation (2). The aspect factor is defined as:

$$r_j(C) = \min \frac{ar(C)}{ar_j}, \frac{ar_j}{ar(C)}. \tag{16}$$

The value of $r_j(C)$ is always smaller than 1, and decreases as the correspondence between the two aspect ratios decreases. This means that the match of an image with a given class is greater when the aspect ratios are more similar, as intuitively expected. This new factor increases recognition success.

Considering the normalised membership functions defined in equation (14), the normalised similarity values in equation (15) and the aspect factor proposed in this paper and defined in equation (16), the test character is classified as belonging to the character group identified by the j^* index, which is defined as:

$$j^* = \arg\max_j \frac{\sum_{i=1}^{N_c} v_{ij} S'_{ij}}{\sum_{i=1}^{N_c} v_{ij}} r_j(C). \tag{17}$$

Summations for j^* are performed over all orientations and each single character group. The test character is finally classified as belonging to the character group identified by the j^* index.

3.4 Recogniser algorithm

The general execution flow of the character recognition algorithm proposed in this paper is summarised next.

1 For each character group, extract features
   - apply Gabor filters to every sample, as defined in equations (1) and (2)
   - compute the average aspect ratio for the character images using (3)
   - compose images per feature orientation.

2 For each character group, perform training
   - perform two-level binarisation using the thresholds in equations (5) and (6)
   - find feature-binding rectangles
   - generate fuzzy membership functions using equations (7) and (8).

3 Perform classification of unknown images
   - extract features as in training, see Steps (1) and (2)
   - compute the weights to each pixel, as defined in equation (12)
   - compute the similarity matrix equation (11)
   - determine the aspect factor introduced in equation (16)
   - determine the most likely classification of a new character in a character group using equation (17).

4 Results

This section presents the character recognition results using the developed fuzzy recogniser. Owing to resource and acquisition limitations, testing could not be performed on an entire book; so, large representative test sets were sought so that the final results can be considered realistic and reliably convey the application performance in an actual common usage environment.

4.1 Training

In general, OCR and OCR-related algorithms require a database of classified character images, for training of the corresponding pattern recognition system. To construct an adequate database, a set of eight pages from the book (de Lião, 1608) was chosen. As a whole, this set contains more than 6400 characters. The FineReader OCR engine was used directly to provide an initial classification of these characters. This automatic classification is not without error (misclassification rate is around 15%), so a manual correction procedure is needed and is performed using a specialised software utility developed by the authors.

Not all characters are used to form the database, however. The characters considered for inclusion are all alphabetic characters belonging to a single typeset, including those with a visual appearance unlike that of their modern typographical equivalent. The latter is classified as the corresponding modern character. A clear example is the elongated *S*, as it was shown in Figure 2. These characters are given a specific class, since mixing different versions of matching characters is unlikely to yield satisfactory results. Accented versions of characters are classified separately as well. Numeric characters will be excluded, because standard OCR techniques already perform well in their recognition. Non-alphanumeric characters are also disregarded, since they do not concern the major objectives of this paper, centred mostly on allowing word search and recognition.

Relative character frequency is also to be taken into account. Some characters occur a lot more frequently than others and some are more often misrecognised than the rest. These facts should be considered to determine the sample size for each character. Frequent characters are more important than infrequent ones. On the other hand, characters that are already successfully recognised most of the times using standard techniques are not given as much attention as other, more problematic cases.

Finally, because automatic character segmentation is not always successful, separate characters are sometimes joined as one, and single characters are occasionally split. Consistently, joined characters can be treated as a group. In some cases, successful segmentation would actually be impossible using bounding rectangles, owing to character shape and positioning.

Summarising, a character database containing 1980 characters from eight pages of the book (de Lião, 1608) was used to perform training of the fuzzy recogniser. The characters have been classified manually. Recognition is possible once the training structure has been generated.

4.2 Testing

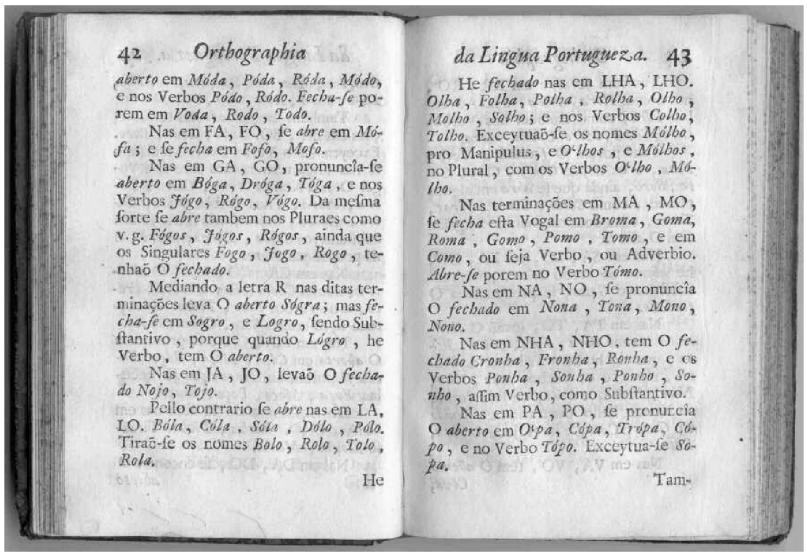
Large representative test sets were sought so that the final results can be considered realistic and reliably convey the application performance in an actual common usage environment. Verifying the results achieved in the recognition tests requires a very time-consuming and thorough manual check of the output. Naturally, the tests actually executed were selected carefully within practical limits in order to be representative. Therefore, three different books have been used for validation purposes; see de Lião (1606) and de Vera (1631a, 1631b).

The book (de Lião, 1606) describes the origin of the Portuguese language. Eight pages were selected, containing 1580 characters. This is a relatively small test set. The second test set consists of 20 pages from the book (de Vera, 1631b). It was acquired

with various paper and printing problems, namely skewing and paper see-through, with both non-italic and italic text. It contains 1886 words consisting in 8034 characters, which is a significantly large test set. The source book (de Vera, 1631b) concerns Portuguese language orthography, providing a large variety of characters and formatting properties. Two pages samples from this book are depicted in Figure 9. Finally, 12 pages of the book (de Vera, 1631a) are also used as validation set. This book discusses the features of the Portuguese language and its similarities with other Latin languages. This book has diverse typesets and several printing problems, even though scanning quality is quite high. This test set has 1590 words.

Per-character results for this test set are summarised in Table 1, where the fuzzy recogniser proposed in this paper is compared to the FineReader engine (ABBYY Software House, 2006), one of the best commercial packages for OCR. The fuzzy recogniser, however, not only has the classification with the highest rate of recognition, but also as second, third, etc. Table 1 presents in the columns concerning the fuzzy recogniser the character that was classified as the highest rate character using equation (17). The second column depicts the percentage that the character was classified as in the top two rating characters from *j*, and finally column 3 indicates the percentage that the character was classified as in the top three rating characters.

Figure 9 Two pages of the validation book



42 Orthographia

aberto em *Móda*, *Póda*, *Róda*, *Módo*, e nos Verbos *Pódo*, *Ródo*. *Fecha-se* porem em *Voda*, *Rodo*, *Todo*.

Nas em FA, FO, fe *abre* em *Mófa*; e fe *fecha* em *Fofo*, *Mofo*.

Nas em GA, GO, pronuncia-fe *aberto* em *Bóga*, *Dróga*, *Tóga*, e nos Verbos *Jógo*, *Rógo*, *Vógo*. Da meſma ſorte fe *abre* tambem nos Pluraes como v. g. *Fógos*, *Jógos*, *Rógos*, ainda que os Singulares *Fogo*, *Jogo*, *Rogo*, tenhaõ O *fechado*.

Mediando a letra R nas ditas terminações leva O *aberto Sógra*; mas *fecha-fe* em *Sogro*, e *Logro*, ſendo Subſtantivo, porque quando *Lógro*, he Verbo, tem O *aberto*.

Nas em JA, JO, levaõ O *fechado Nojo*, *Tojo*.

Pello contrario ſe *abre* nas em LA, LO. *Bóla*, *Cóla*, *Sóla*, *Dólo*, *Pólo*. Tiraõ-ſe os nomes *Bolo*, *Rolo*, *Tolo*, *Rola*.

He

da Lingua Portugueza. 43

He *fechado* nas em LHA, LHO. *Olha*, *Folha*, *Polha*, *Rolha*, *Olho*, *Molho*, *Solho*; e nos Verbos *Colho*, *Tolho*. Exceytuaõ-ſe os nomes *Mólho*, pro Manipulus, e *Olhos*, e *Mólhos*, no Plural, com os Verbos *Olho*, *Mólho*.

Nas terminações em MA, MO, ſe *fecha* eſta Vogal em *Broma*, *Goma*, *Roma*, *Gomo*, *Pomo*, *Tomo*, e em *Como*, ou ſeja Verbo, ou Adverbio. *Abre-ſe* porem no Verbo *Tómo*.

Nas em NA, NO, ſe pronuncia O *fechado* em *Nona*, *Tona*, *Mono*, *Nono*.

Nas em NHA, NHO, tem O *fechado Cronha*, *Fronha*, *Ronha*, e os Verbos *Ponha*, *Sonha*, *Ponho*, *Sonho*, aſſim Verbo, como Subſtantivo.

Nas em PA, PO, ſe pronuncia O *aberto* em *Ópa*, *Cópa*, *Tópa*, *Cópo*, e no Verbo *Tópo*. Exceytua-ſe *Sópa*.

Tam-

Source: de Vera (1631b)

Table 1 Per-character success rates (in percentage)

	FineReader engine	*Fuzzy recogniser*		
		1	*2*	*3*
Test set in (de Lião, 1606)	87.5	88.9	96.6	97.9
Test set in de Vera, A.F. (1631b)	86.9	88.2	89.2	89.7
Test set in de Vera, A.F. (1631a)	79.2	80.9	83.9	85.2

Table 1 shows that both systems successfully classified the three different test sets. The improvement introduced by the fuzzy recogniser is slight, although consistent using only the highest rated character. Using also the other rates, the fuzzy recogniser clearly outperforms the FineReader engine. Many errors occur because of printing defects and strong similarity between certain key characters. The test set in (de Lião, 1606) is the one that improves most using the second and the third classified characters. It goes up to 97.9%, which is a remarkable figure. The other two test sets from books (de Vera, 1631a, 1631b) increase the rate of recognition, but not in such a remarkable way. The editors of these two books were different, and the characters changed slightly. Therefore, the results recommend to include the examples of the same editor when recognising a book, in order to increase the recognition rate. However, the recognition is still quite high, which would enable word recognition.

The two test sets in de Vera (1631a, 1631b) were used to evaluate the standard FineReader recognition and the fuzzy recogniser output. Table 2 shows the success rate for each of these cases in the two test sets. Recognition output was analysed on a word basis; any word with at least one misclassified character is considered wrong; checking is case-insensitive and graphical accents are ignored.

Table 2 Per-word success rates (in percentage)

System	*Test set in de Vera* (*1631b*)	*Test set in de Vera* (*1631a*)
FineReader engine	62.9	34.6
Fuzzy recogniser	64.5	35.9

Per-word results can be considered unfair towards the FineReader and fuzzy recognisers, because these are character-based and not word-based. Most wrong words had few incorrect characters, explaining why per-character success rates are higher than per-word success rates. Note however that the fuzzy recogniser is again consistently better than the FineReader. A possible solution to improve these results would be the use of a dictionary, such as the GNU ASpell created by Kevin Atkinson, which is a free open-source spell checker that can be used as a stand-alone application or as a linkable software library (Atkinson, 2006). It is designed to retain some compatibility with ISpell, the standard UNIX spell checker, while offering superior performance and several technical improvements. A large set of languages, including Portuguese, is supported through dictionary modules loaded dynamically in run-time. ASpell is a fast spell checker, but it can also suggest a rich set of words as possible corrections; it was conceived specifically with this ability in mind. Suggestions are used to improve recognition success rate through spell correction and to assist in post-recognition word splitting when combined with effective word distance metrics. Note however that this or other dictionaries are developed for modern Portuguese, not for old spellings of the language. Therefore, the difference in words decreases the word recognition rate, and cannot be validated. Moreover, books from different centuries have different spellings. This is another reason for using character recognition instead of word recognition. Therefore, this paper did not investigate further word recognition of old documents.

In terms of speed of the proposed algorithm, the fuzzy OCR system completed the recognition of 20 pages in 198s (about 10s per page) in a Pentium 4 running at 2.5 GHz with 1 GB of RAM.

5 Conclusions

This paper proposed a character recogniser based on fuzzy pattern recognition. The system was designed to recognise old printed documents, containing several defects. Improvements of recognition results were noticeable with the fuzzy recogniser module. The fuzzy system achieved a success rate that is better than a mature commercial software package. Therefore, the results confirm that using expert systems in image-based decision problems is a good solution to deal with images suffering from uncertainty problems, as e.g., aging problems in old printed documents, as discussed in this paper.

Further work can include the development of an automatic parameter-adjustment system based on measurable properties of the documents being processed, the introduction of more accurate heuristics and the development of an ancient word dictionary for spell checking.

Acknowledgements

This work was partly supported by: the "Programa de Financiamento Plurianual de Unidades de I&D (POCTI), do Quadro Comunitario de Apoio III"; the FCT project POSI/ SRI/41201/2001; "Programa do FSE-UE, PRODEP III, Quadro Comunitario de Apoio III"; and program FEDER.

We also wish to express our thanks to the Portuguese National Library (Biblioteca Nacional), especially to Professor José Borbinha, for their continuous support.

References

ABBYY OCR (2005) *Awards and Reviews*, http://www.abbyy.com/company/?param=1778.

ABBYY Software House (2006) *ABBYY FineReader*, http://www.abbyy.com.

Atkinson, K. (2006) *Gnu Aspell Homepage*, http://aspell.net, GNU Project.

Buse, R., Liu, Z.Q. and Bezdek, J. (2001) 'Word recognition using fuzzy logic', *IEEE Transactions on Fuzzy Systems*, Vol. 10, No. 1, February, pp.65–76.

Buse, R., Liu, Z.Q. and Caelli, T. (1997) 'A structural and relational approach to handwritten word recognition', *IEEE Transactions on Systems, Man and Cybernetics, PartB: Cybernetics*, Vol. 27, No. 5, October, pp.847–861.

Chen, S.J., Hwang, C.L. and Hwang, F.P. (1992) *Fuzzy Multiple Attribute Decision Making, Methods and Applications*, Springer-Verlag, Berlin, Germany.

de Lião, D.N. (1606) *Origem da Lingoa Portvgvesa*, Available at Biblioteca Nacional (Portuguese National Library), Lisbon, Portugal, 17th Century.

de Lião, D.N. (1608) *Orthographia da Lingoa Portvgvesa*, Available at Biblioteca Nacional (Portuguese National Library), Lisbon, Portugal, 17th Century.

de Vera, A.F. (1631a) *Breves Lovvores da Lingva Portvgvesa, com Notaveis Exemplos da Muita Semelhana, Que Tem com a Lingua Latina*, Available at Biblioteca Nacional (Portuguese National Library), Lisbon, Portugal, 17th Century.

de Vera, A.F. (1631b) *Orthographia ou Modo Para Escrever Certo na Lingua Portuguesa*, Available at Biblioteca Nacional (Portuguese National Library), Lisbon, Portugal, 17th Century.

Foley, J.D., van Dam, A., Feiner, S.K. and Hughes, J.F. (Eds.) (1990) *Computer Graphics: Principles and Practice in C*, 2nd ed., Addison-Wesley, Massachussets, USA.

Gabor, D. (1946) 'Theory of communications', *J. Inst. Electr. Eng*., Vol. 93, pp.429–457.

Hwang, C.L. and Yoon, K. (1981) *Multiple Attribute Decision Making, Methods and Applications, A State-of-the-Art-Survey*, Springer-Verlag, Berlin, Germany.

Jain, L.C. and Lazzerini, B. (Eds.) (1999) *Knowledge-Based Intelligent Techniques in Character Recognition*, CRC Press, Boca Raton, Florida.

Mori, S., Nishida, H. and Yamada, H. (1999) *Optical Character Recognition*, Wiley Interscience, New York.

Otsu, N. (1979) 'A threshold selection method from gray level histograms', *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 9, No. 1, January, pp.62–66.

Parker, J.R. (Ed.) (1998) *Algorithms for Image Processing and Computer Vision*, Wiley & Sons, New York, USA.

Sousa, J.M.C. and Kaymak, U. (2002) *Fuzzy Decision Making in Modeling and Control*, World Scientific Pub. Co., Singapore.

Toussaint, G. (1983) 'Solving geometric problems with the rotating calipers', *Proc. IEEE Mediterranean Electrotechnical Conference, MELECON'83*, IEEE, Athens, Greece, May, pp.A10.02/1–4.