Character Extraction from Documents using Wavelet Maxima

Wen L. Hwang and Fu Chang

Institute of Information Science

Academia Sinica, Taiwan, R.O.C.

E-mail: whwang@iis.sinica.edu.tw

Abstract

The extraction of character images is an important front-end processing task in optical character recognition (OCR) and other applications. This process is extremely important because OCR applications usually extract salient features and process them. The existence of noise not only destroys features of characters, but also introduces unwanted features. We propose a new algorithm which removes unwanted background noise from a textual image. Our algorithm is based on the observation that the magnitude of the intensity variation of character boundaries differs from that of noise at various scales of their wavelet transform. Therefore, most of the edges corresponding to the character boundaries at each scale can be extracted using a thresholding method. The internal region of a character is determined by a voting procedure, which uses the arguments of the remaining edges. The interior of the recovered characters is solid, containing no holes. The recovered characters tend to become fattened because of the smoothness applied in the calculation of the wavelet transform. To obtain a quality restoration of the character image, the precise locations of characters in the original image are then estimated using a Bayesian criterion. We also present some experimental results that suggest the effectiveness of our

method.

1 Introduction

This paper presents an algorithm for the extraction of character images from possibly polluted textual documents. It differs from earlier techniques in that the edges of a textual document are extracted using the modulus local maxima of the wavelet transform. Most of the edges of the character boundaries are distinguished from that of noise at each scale by a simple thresholding process. After suppression of most of the edges of noise at each scale, the interior of characters is recovered by a voting procedure which uses the arguments of edges. The characters recovered after such a procedure have a tendency to become broadened. To restore the precise locations of characters, a statistical decision based on a Bayes criterion is applied.

The extraction of character images is an important front-end process in Optical Character Recognition (OCR) applications. Its effect is to produce character images by suppressing unwanted patterns from a textual document. After the extraction of characters, applications such as character recognition, textual transmission, document block segmentation and identification can be carried out.

Many algorithms [6][7][10][3][8] have been proposed to restore character images in noisy environments. A variety of them [6][8] apply thresholding techniques directly to the histogram of the gray-scaled image. The histogram, in the ideal case, has a valley between two peaks, representing the aggregations of character pixels and background pixels, respectively. A threshold is found within the valley to distinguish characters from their background. These algorithms are simple and fast. However, their discrimination of fine details along character boundaries is not very accurate in many applications. Due to distortions caused by scanners and other devices, the histogram of a gray-scaled image usually is far more complex than simply a combination of several peaks. An example is give in Figure 1, where the gray level histogram is approximately uniform. Another class of algorithms [10][7] selects thresholds which vary with the local properties of the textual image. The approach compares the gray value of a given pixel with the average gray values of its neighboring pixels. A typical example is the dynamic threshold algorithm proposed by White and Rohrer [10]. The decision of a pixel to be black or white is determined by a threshold level which changes according to changes of the gray levels of the input stream. In their settings, the threshold calculation is a first-order difference equation with nonlinear coefficients. The second algorithm of White and Rohrer [10], referred to by them as the integrated function algorithm, labels edge pixels in raw images and determines the dark and light sides with the help of the Laplacian operator.

The main idea behind this paper is the use of sharp intensity variation (called edges hereafter) and the local thresholding technique in the extraction of characters from a textual document. Noise reduction algorithms usually require advance knowledge of the type of noise to be suppressed. However, in general, it is unlikely to have a procedure which can characterize automatically what the unwanted patterns are in a textual image. Moreover, observable noise in a textual image is derived from diverse sources, each affecting the image in a different way. There are, for example, salt and pepper noise, scanner noise introduced due to nonuniform illumination, and smeary spots in a document exposed to water or grease. These types of noise bear no similarities and are impossible to enumerate and characterize. Thus, some *a priori* knowledge usually is required for the character image to be preserved in a character extraction process. We assume, in this paper, that documents are gray-scaled images with no pictorial contents in which pictures have been removed. Also, a character is a collection of connected components of dark pixels.

It has been shown that if a wavelet is chosen as the partial derivative of a smoothing function, then edges of an image at different scales, as defined by Canny [1], will correspond to the local maxima of the modulus of the wavelet transform calculated along the gradient direction (called the wavelet maxima) [5]. We thus choose the wavelet, designed by Mallat and Zhong [5], which is the partial derivative of a cubic spline smoothing function, to build the wavelet transform of a textual image. The notions of edge and wavelet maximum are equivalent hereafter in this paper. Edges corresponding to character boundaries at each scale are preserved by thresholding the modulus of the wavelet maxima [2]. It is known that, from the analysis of singularities [4], the modulus of the wavelet maxima corresponding to character boundaries and that corresponding to noise varies differently along different scales. Hence, effective threshold levels, which distinguish character boundaries from noise, can be calculated at the scales where the average modulus of noise and character boundaries varies. Most of the noise can then be removed while most of the edges of the character boundaries are kept at these scales. For the purpose of character extraction, it is also important to find the interior of each character. To this end, we use the arguments of edges to decide to which region, either the interior or the exterior of a character, a pixel should belong. For any given pixel and any given line passing through the pixel, we can associate the pixel with two edge points on the line where there is no other edge point within the interval covered by the pair of edge points (see Figure 3). Given the two edge points associated to a pixel, we can then deduce from their arguments if the pixel is within a character (also see Figure 3). An image pixel has four lines passing through it. Hence, there are four pairs of edges associated to the pixel. In a noisy environment, the region to which a pixel belongs can not be determined reliably simply from one pair of edges (called an edge-pair for short). Thus, a voting procedure from all the orientations of the edge-pairs is used in determining the interior of a character. The characters recovered using this voting procedure will then contain no background spots within their interiors. However, locations of recovered edges are not precise because the wavelet transform tends to broaden character boundaries. As a remedy, a statistical decision based on the Bayes test is used to estimate the precise locations of characters in the original textual image.

The rest of this paper is arranged as follows. In Section 2, the background of the dyadic wavelet transform and its local maxima is reviewed. In Section 3, we present algorithms which use the wavelet local maxima to distinguish character contours from their backgrounds at each scale. Methods for finding the interiors of characters from their boundaries are also described in this section. Section 4 describes the use of the Bayes criterion to determine the precise locations of characters. Section 5 presents the algorithm in detail. We also discuss issues related to the selection of free parameters. Some experimental results are given in last section.

2 The Dyadic Wavelet Analysis of Images

This section is devoted to describing the specific wavelet transform used in this paper. For character image extraction, we will restrict ourselves to use the dyadic wavelet transform of images [5]. That is to say, the wavelet transform of a single image is applied only to the dyadic scale 2^j.

We will review, in the following, the dyadic wavelet transform of an image and its modulus local maxima. We call a two-dimensional smoothing function any function whose double integral is nonzero. Wavelets that are, respectively, the partial derivatives along x and y of a two-dimensional smoothing function $\theta(x,y)$ are defined as:

$$\psi^1(x,y) = \frac{\partial\theta(x,y)}{\partial x} \quad \text{and} \quad \psi^2(x,y) = \frac{\partial\theta(x,y)}{\partial y}.$$

Let $\psi^1_{2^j}(x,y) = (1/2^j)^2\psi^1(x/2^j, y/2^j)$ and $\psi^2_{2^j}(x,y) = (1/2^j)^2\psi^2(x/2^j, y/2^j)$. For any function $f(x,y) \in \mathbf{L}^2(\mathbf{R}^2)$, the dyadic wavelet transform defined with respect to $\psi^1(x,y)$ and $\psi^2(x,y)$, at the scale 2^j, has two components:

$$\mathcal{W}^1 f(2^j,x,y) = f * \psi^1_{2^j}(x,y),$$

and

$$\mathcal{W}^2 f(2^j,x,y) = f * \psi^2_{2^j}(x,y),$$

where the symbol $*$ is the convolution. One can show that

$$\begin{aligned} \begin{pmatrix} \mathcal{W}^1 f(2^j,x,y) \\ \mathcal{W}^2 f(2^j,x,y) \end{pmatrix} &= 2^j \begin{pmatrix} \frac{\partial}{\partial x}(f * \theta_{2^j})(x,y) \\ \frac{\partial}{\partial y}(f * \theta_{2^j})(x,y) \end{pmatrix} && (1) \\ &= 2^j \vec{\bigtriangledown}(f * \theta_{2^j})(x,y). && (2) \end{aligned}$$

Hence, the two components of the wavelet transform at the scale 2^j are proportional to the gradient vector of $f(x,y)$ smoothed by $\theta_{2^j}(x,y)$. The wavelet transform at the scale 2^j is nothing but the gradient of this smoothed version of the original image. The orientation of the gradient vector indicates the direction where the partial derivative of $f(x,y)$ has an absolute value which is maximum. It is the direction where $f(x,y)$ has the sharpest variation. We shall use the modulus $\mathcal{M}f(2^j,x,y)$ and the angle $\mathcal{A}f(2^j,x,y)$ of this vector:

$$\mathcal{M}f(2^j,x,y) = \sqrt{|\mathcal{W}^1 f(2^j,x,y)|^2 + |\mathcal{W}^2 f(2^j,x,y)|^2},$$
$$\mathcal{A}f(2^j,x,y) = \tan^{-1}(\frac{\mathcal{W}^1 f(2^j,x,y)}{\mathcal{W}^2 f(2^j,x,y)}).$$

At the scale 2^j, the modulus maxima of the dyadic wavelet transform are defined as the points (x,y) where the modulus image $\mathcal{M}f(2^j,x,y)$ is a local maximum along the gradient direction given by $\mathcal{A}f(2^j,x,y)$. This is the same as the definition of an edge point given by Canny [1]. Therefore, in the following, the notations for edges and the wavelet transform modulus local maxima are equivalent in our paper.

Although fine scale and coarse scale are a relative terms, conventionally, fine scale refers to a smaller dyadic scale, usually 2^1 or 2^2, and coarse scale to a larger dyadic scale. The locations of edges at a fine scale are more precise than those at a coarse scale because the wavelet at a finer scale has less support than does that at a coarser scale in calculation of edge point.

3 Character Image Extraction

3.1 Extraction of Character Boundaries

The boundaries of characters appear as edges in a raw document image. The gray values of a character vary smoothly along its contours but sharply in the direction perpendicular to the contours. Hence, character boundaries appear as smooth curves along the contour and as singular edge points in the direction perpendicular to the contour. As discussed above, the edges of character boundaries are detected by the modulus of the wavelet transform. These edges have gradients pointing in directions perpendicular to the boundary contours.

From the analysis of singularities [4], the modulus of wavelet maxima will decay along scales, with a rate depending to the type of singularity. In general, it is rather difficult to calculate exactly the decay of the modulus of wavelet maxima of the character boundaries and noise along scales. However, since the edges of character boundaries and noise usually have different types of singularities, we can still discriminate the edges of characters from those of noise according their modulus of wavelet maxima. Following a similar work in [2], we use a simple thresholding method for this task. A threshold at each scale is selected such that most of the edges of noise are removed while most of the character boundaries are preserved. We select the 100α-th percentile, say $\pi_\alpha(2^j)$, of the modulus of edges as the threshold at scale 2^j, where α is between 0 and 1. Next, we determine the edges at (x, y) with modulus above this percentile threshold. In other words, we select the set of edges at (x, y), where

$$\{(2^j, x, y)_{j=1,2,\ldots J} : \mathcal{M}f(2^j, x, y) > \pi_\alpha(2^j)\},$$

which we call the contour map. At the top of Figure 2 is shown an original textual image, taken from a piece of aged newspaper. The figures in the middle and at the bottom are the modulus maxima of this noisy image, calculated at the scales 2^1 and 2^2, respectively. At the finest scale, noise dominates the characters. The shapes of the characters can hardly be discriminated. At a coarser scale, 2^2 in this example, the character boundaries can be discriminated, which means that the properties of the characters dominate those of the noise more and more as the scale increases.

In practice, it is impossible to derive an analytical threshold at each scale: Although the edges of character boundaries can be modeled as step edges at finer scales, edge types may vary and become intractable when the modulus of the edge is blended with the its neighboring pixels as the scale increases. Also, most of the character boundaries at finer scales are dominated by noise (see the middle of Figure 2), this makes derivation of analytical threshold even more difficult. Thus, the values of thresholds can only be obtained empirically. Fortunately, the thresholding techniques in [6] can be applied at each scale to discriminate many character boundaries since the moduli of noise and character boundaries aggregate at separate values and, thus, can be delineated using a thresholding method. Shown in the middle and at the bottom of Figure 5 are the contour maps obtained after the suppression of irregular variations of noise by applying the 50th percentile and 70th percentile thresholds of the modulus at the scales 2^1 and 2^2, respectively.

3.2 The Interior of Characters

After applying thresholding on the edge map at each scale, the remaining edges are more or less those pixels that lie on character contours (thus, they are called contour maps.) So far, we have only extracted the boundaries of characters. For the purpose of extracting image characters, it is also impossible to determine the interior of characters. Our next step is to label the pixels within a character.

We use the coordinate system where x increases in value if we move toward the right, and y increases in value if we move downward in an image (see Figure 3). In a typical 8-neighborhood image system, there are four lines passing through a pixel (x, y). They are represented as: $\{(x-1, y)(x+1, y)\}$, $\{(x, y-1)(x, y+1)\}$, $\{(x-1, y-1)(x+1, y+1)\}$, and $\{(x-1, y+1)(x+1, y-1)\}$. Recall that we have assumed that the character has dark gray values. This will help us in determining the interiors of characters. Our idea is illustrated in Figure 3: if a pixel p is within a character, then the four lines passing through the pixel will intersect the boundaries of the enclosing character. Each line intersects the character's boundaries at least twice. We can associate to p the two edge points on the line such that there is no other edge point within the interval of the two edge points (called the edge-pair of the pixel p). In Figure 3, four edge-pairs, $(A-a)$, $(B-b)$, $(C-c)$, and $(D-d)$, are indicated. Our algorithm for determining the interior of a character is based on the observation that if a pixel is located within a character, then the arguments (orientations) of the edge-pairs should agree with the *a priori* knowledge that a character has a darker gray value than its background. The left side of Figure 3 shows the arguments of the four edge-pairs of the pixel p if the character pixels have darker gray values, while on the right side of the figure are the arguments if the background pixels have darker gray values.

Although one can determine whether a pixel is within a character simply from the arguments of any edge-pair of the pixel, in practice, we need a robust determination due to the existence of noise and errors in edge detection. We, thus, adopt a voting strategy in robust determination: A consensus from the majority of the edge-pairs of a pixel is required in determining whether the pixel is within a character. Each pixel is associated with a score which consists of the votes from its edge-pairs. We add 1 to the score of a pixel if an edge-pair of the pixel confirms that the pixel is located within a character. Ideally, there are 4 votes for a pixel inside a character. However, errors, such as broken character boundaries generated in the process of detecting edges and suppressing noise, will cause this number to fall short of 4. Thus, we ccount a voted majority to determine whether a pixel is inside a character. This voting procedure is effective: if a pixel is within a character, there is a high probability that it will have a high score. On the other hand, it is unlikely that a pixel lying outside a character will have a high score.

Let $\mathcal{Y}(2^j, x, y)$ be the number of votes for the pixel (x, y) to be located within a character at the scale 2^j. We then sum over the dyadic scales to collect all the votes $\mathcal{V}(x, y)$ at (x, y):

$$\mathcal{V}(x, y) = \sum_{j=1}^{J} \mathcal{Y}(2^j, x, y).$$

Given that the value of $\mathcal{Y}(2^j, x, y)$ is at most 4 at the scale 2^j, the value of $\mathcal{V}(x, y)$ is at most $4J$. The larger the value $\mathcal{V}(x, y)$ is, the higher is the probability that the pixel (x, y) will be within a character. We select from $\mathcal{V}(x, y)$ the set of pixels having values larger than a given threshold π_v to be within a character:

$$\{(x, y) : \mathcal{V}(x, y) > \pi_v\}.$$

The method used to determine the interior of a character, just described, not only keeps the components of a character, but also preserves some dark spots which originate from noises. We then assume that a character is a collection of strokes with more than π_s dark pixels. Then, we remove the small dark spots by comparing their sizes with the number π_s. We call the resultant image a character map since we have recovered not only the character boundaries but also the interior of the characters.

Figure 4 shows the character map of the original textual document shown at the top of Figure 2. One can see from the figure that errors in the arguments of edges and in discriminating character edges cause two edge points in different characters to vote for the pixels between them. This yields a chord between the two characters. In the meantime, because the smoothing function $\theta_{2^j}(x, y)$, whose support is proportional to 2^j, is applied in the wavelet transform, the contours of characters tend to broaden. Thus, the positions of the character boundaries at a fine scale are more precise than those at a coarse scale because less smoothing is applied at a fine scale. The broadened characters appear as a blurred image at lower resolution. Thus, it is important to estimate the precise locations of character boundaries.

3.3 The Estimation of Character Locations using Bayes Test

In the following, we will consider the binary hypothesis: whether a pixel corresponds to a character image or to the background. We shall confine our decision of the location of a pixel in the original document as either H_1, meaning that the pixel corresponds to a character image, or H_0, meaning that the pixel corresponds to the background. Thus for a given pixel, one of four possible outcomes will occur:

- H_0 is true, H_0 is chosen,
- H_0 is true, H_1 is chosen,
- H_1 is true, H_0 is chosen, and
- H_1 is true, H_1 is chosen.

The first and last alternatives correspond to correct choices. The second and third alternatives correspond to wrong choices. The source output of a pixel is governed by the *a priori* probability assignment of H_1 and H_0, which are denoted by P_1 and P_0, respectively. The costs of the four outcomes are denoted as C_{00}, C_{01}, C_{10}, and C_{11}, respectively. The first subscript indicates whether the hypothesis is true or false, and the second subscript indicates whether that the hypothesis is chosen or not. The Bayes test is designed so that the likelihood threshold Γ is chosen so as to minimize the following expected cost as the risk $\mathcal{R}$:

$$\begin{aligned}
\mathcal{R} \;=\;& P_0\, C_{00}\, P(H_0 \;\; chosen|H_0 \;\; true)+ \\
& P_0\, C_{01}\, P(H_1 \;\; chosen|H_0 \;\; true)+ \\
& P_1\, C_{10}\, P(H_0 \;\; chosen|H_1 \;\; true)+ \\
& P_1\, C_{11}\, P(H_1 \;\; chosen|H_1 \;\; true).
\end{aligned}$$

According to the Bayes criterion for the minimization of the risk $\mathcal{R}$, the region (either character image or background) to which a pixel is assigned will depend on the prior probabilities P_0 and P_1 and the four costs functions C_{ij}, where both $i, j \in \{0, 1\}$. In general, these prior probabilities and cost functions are unknown. In the Bayes decision, these unknown terms are usually collected as a new term, call the the likelihood threshold Γ. For a detailed analysis of how the likelihood threshold Γ is obtained, the reader can refer to [9]. The likelihood Γ is:

$$\Gamma = \frac{P_{r|H_1}(R|H_1)}{P_{r|H_0}(R|H_0)},$$

where $P_{r|H_1}(R|H_1)$ and $P_{r|H_0}(R|H_0)$ are the probability densities of an observation R, given that a pixel is within a character (H_1 hypothesis) or within the background (H_0 hypothesis), respectively. The way to determine the likelihood Γ is to compute the ratio of these two probabilities densities. Because the natural logarithm is a monotone function, we can determine the threshold from the log likelihood $\ln \Gamma$ equally well :

$$\ln \Gamma = \ln P_{r|H_1}(R|H_1) - \ln P_{r|H_0}(R|H_0).$$

We assign a pixel to the H_1 hypothesis if the log likelihood threshold $\ln \Gamma$ is greater than 0. Otherwise, the pixel is assigned to the H_0 hypothesis. Although the exact densities $P_{r|H_1}(R|H_1)$ and $P_{r|H_0}(R|H_0)$ are unknown in our applications, we can still estimate their ratio by appropriate assumptions: Let $g(p)$ denote the gray value of p. Given a neighborhood of p, let mc be the median of the gray values of the neighboring pixels of p that are within the character map, and let mb be the median of those that are outside of the character map. The distance between $g(p)$ and mc is an approximation of the measurement of $\ln P_{r|H_1}(R|H_1)$, and, by the same token, the distance between $g(p)$ and mb is an approximation of $\ln P_{r|H_0}(R|H_0)$. The log likelihood is computed as

$$\begin{aligned} \ln \Gamma(p) &= \ln P_{r|H_1}(R|H_1) - \ln P_{r|H_0}(R|H_0) \\ &\approx k\,(|g(p) - mc| - |g(p) - mb|), \end{aligned}$$

which is the sufficient statistics for the log likelihood estimation, where k is a positive number.

We have found that after applying the above hypothesis test to the character map, the character boundaries of the recovered image are more precise. The characters no longer appear fattened or smeared due to the smoothed boundaries. The top of Figure 5 shows the characters extracted after applying the Bayes criterion from the character map in Figure 4.

4 Detailed Implementation and Experimental Results

We will now outline the detailed implementation of the algorithm for extraction of character images. Then, we will discuss issues related to the selection of free parameters and the robustness of these parameters in our algorithm. Finally, we will discuss the performance issue.

1. Compute the wavelet transform up to scale 2^J.

2. Compute the modulus local maxima of the wavelet transform at each scale.

3. For each scale 2^j, where $j = 1, 2, ..J$, we keep only the edges having moduli greater than the 100α-th percentile threshold, where α is between 0 and 1. The resultant image is called a contour map.

4. The interiors of characters are extracted from the contour map using a voting procedure. The more votes a pixel gets, the higher the probability is that the pixel is located in a character. We keep only those pixels which accumulate more than π_v votes.

5. Remove the remaining noisy spots using the assumption that the size of a dark pixel of a component of a character is at least π_s. Hence, these dark spots whose sizes are less than π_s are removed. The resultant image is called a character map.

6. We determine the precise locations of characters using the original image and the character map. A log likelihood threshold, based on Bayes criterion, is defined as

   $$\ln \Gamma(p) = k(|g(p) - mc| - |g(p) - mb|).$$

   The pixel p is determined as being within a character if $\ln \Gamma(p)$ is greater than 0; otherwise, it does not belong to a character.

In the first step, the wavelet transform is computed up to a certain scale, that is, up to 2^2 for all of our experiments. We keep the edges with moduli greater than the 100α-th percentile threshold at each scale. The selection of the percentile depends on the average moduli of the character edges and noise edges. Since the average moduli of the noise and character boundaries vary along the scales, and the percentile varies accordingly. Strictly speaking, there is no analytical way of determining the percentile for each scale. It is recommended the percentile threshold preserves as many edges of characters as possible and, meanwhile, suppress as many edges of noise at each scale. If a percentile is chosen which is too low, many edges of noise will be kept. This noise will influence the later processes, especially the process of determining the interiors of characters. On the other hand, if a percentile is chosen which is too high, many edges of characters will be suppressed. As a consequence, the broken character boundaries and lost parts will, in general, be quite hard to restore. Fortunately, a thresholding technique such as that in [6] can be applied to the modulus of the wavelet local maxima at each scale. In all the experiments we have carried out, we found it appropriate to use approximately the 50-th percentile as a threshold for the scale 2^1 and the 70-th percentile at the scale 2^2.

The next step is to determine the interiors of characters. Recall that a pixel belongs to the interior of a character if it obtains enough votes. A character pixel can obtain at most 4 votes at each scale with a total of $4J$ votes up to the scale 2^J. In our experiments, a threshold π_v is used so that a pixel is determined as being within a character if it has at least $2J$ votes; hence, $\pi_v = 2J$.

According to *a priori* knowledge, a character is composed of components of dark pixels. The sizes of the pixels of each component are at least π_s, which was set to 10 pixels in our experiments. For a 300 dpi scanner, 10 pixels are about 1/30 inch. That is less than most of the pixels of the connected components of characters. Thus, dark components with less than 10 pixels are removed.

In the last step of our algorithm, a Bayes test is applied to determine the region a pixel falls in. We require a neighborhood of size π_n so that enough neighboring pixels are used to calculate the log likelihood threshold. In our algorithm, the neighborhood is defined as a square box with 3600 pixels in total.

We have already discussed ways of selecting thresholds and their robustness in our algorithm. As noted before, noise can be introduced by the scanner. At the top of Figure 1 is shown a portion of a blurred textual image contaminated by a scanner. This image is interesting in the sense that its gray level histogram is approximately uniform except around the pixel value 255 (see the bottom of Figure 1). Thus, the methods which determine the threshold based on the histogram of gray values will fail because it will be impossible to identify peaks in the histogram corresponding to the characters and the background, respectively. However, the character boundaries are still perceived clearly. The restored image is shown in the middle of Figure 1. The unpleasant appearance caused by blurring in the original gray-scaled image has been removed by our process. A portion of an English textual document is shown on the left side of Figure 6, and on the right side of Figure 6 is the result of character extraction from the document. Further experiments were conducted on the slightly degraded image shown at the top of Figure 7. In the middle and at the bottom of this figure are the binarized images which resulted from applying the binarization algorithm provided by the scanner (UMAX Vista Scan S8) and by our algorithm, respectively. The characters in these two images were then input into the same commercial product, MAXReader Pro V2.0, for recognition. The number of characters recognized by MAXReader was 169 for the middle image and 214 for the bottom image out of a total of 225 characters. The results showed that our method improves the recognition rate from 75% to 95%, with an improvement of recognition rate of 20%.

Finally, a comment on the performance of our algorithm: several steps, such as steps 1 and 6, are costly. However, the main purpose of this paper has been to provide the necessary operators that work for a broader class of textural documents. Actually, the cost of the algorithm can be reduced if a different implementation is carried out.

Remark. The approach of the wavelet transform modulus maxima using Mallat and Zhong's wavelet can be regraded as the multi-scale Canny edge detector. However, there is much more the wavelet modulus maxima than the Canny edges because Mallat and Zhong have shown in [5] that a close approximation of the original image can be obtained from the dyadic wavelet modulus maxima from the scales 2^1 to 2^J as well as the coarse scale image at the scale 2^J. However, they used an alternative projection process which is often time-consuming. In this paper, we have not fully explored the advantages of the wavelet modulus representation in noisy reduction as in [4] and [2]. In character extraction, the locations of character boundaries, the location of a pixel whether or not it is within a character, and the efficiency of processing are important factors while the precise reconstruction of a “denoised” image from the wavelet modulus maxima is not required. Thus, we have adopted a faster but less precise way of recovering character images.

5 Conclusion

This paper has described a new algorithm for the extraction of a character image from a textual image. We have used the modulus of wavelet transform to detect character boundaries. According to the analysis of singularities, character edges and noise are separated by means of the choice of threshold for the modulus size. Using our algorithm, the interiors of characters can be delineated robustly when the edges of characters have been located. Moreover, more precise estimation of character boundaries is obtained by using a statistical decision procedure based on the Bayes test. Characters are recovered without smearing of separate components. These procedures have been applied to several textual images and proved to be effective.

Acknowledgement: The authors would like to thank Mr. Tzu-Ming Tang for many valuable discussions and for his help in preparing textual documents.

References

[1] J. Canny (1986): A computational approach to edge detection. *IEEE Trans. Pattern Anal. Machine Intell.* **vol.8**, 679–698.

[2] R. Carmona, W.L. Hwang, and R. Frostig (1995): Wavelet Analysis for Brain-Function Imaging. *IEEE Trans. Medical Imaging.*

[3] S. Liano and M. Ahmadi (1994): A Morphological Approach to Text String Extraction from Regular Periodic Overlapping Text/Background Images. *CVGIP: Graph Models and Image Processing* **vol. 56, No.4**, 402-413.

[4] S. Mallat and W.L. Hwang (1992): Singularities Detection and Processing with Wavelets. *IEEE Trans. Info. Theory,* **38#2**, 617–643.

[5] S. Mallat and S. Zhong (March 1992): Characterization of Signals from Multiscale Edges. *IEEE Trans. Pattern Anal. Machine Intel.* **vol. 38, No.2**.

[6] N. Otsu (1979): A Threshold Selection Method from Gray-Level Histograms. *IEEE trans. Sys. man, and cyber.* **vol. 9, No. 1**, 62-66.

[7] T. Pavlidis (1993): Threshold Selection Using Second Derivatives of The Gray Scale Image. *IEEE.*

[8] W. H. Tsai (1994): Moment-Preserving Techniques and Applications – A tutorial. *Conference on CVGIP, Taiwan.*

[9] H.L. Van Trees (1968): *Detection, Estimation, and Modulation Theory, part* **1**.

[10] J. M. White and G. D. Rohrer (1983): Image Thresholding for Optical Character Recognition and Other Applications Requiring Character Image Extraction. *IBM J. Develop.* **vol. 27, No. 4**.

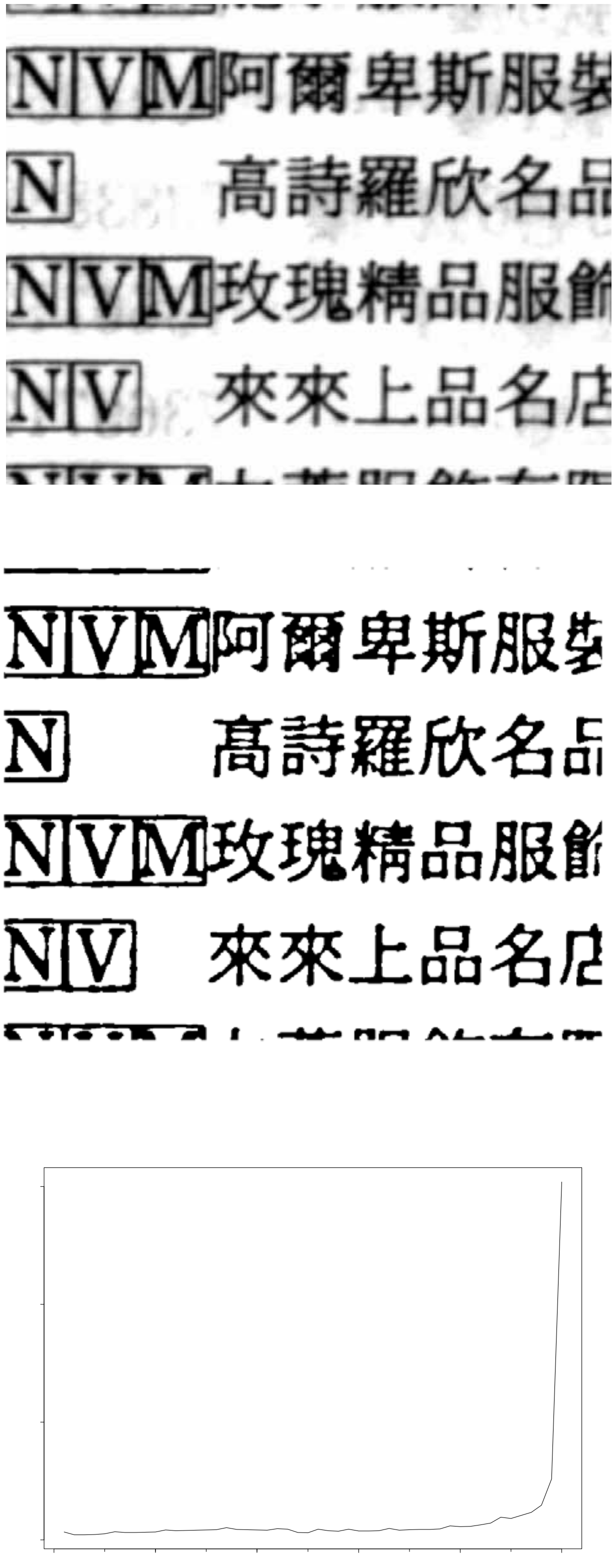
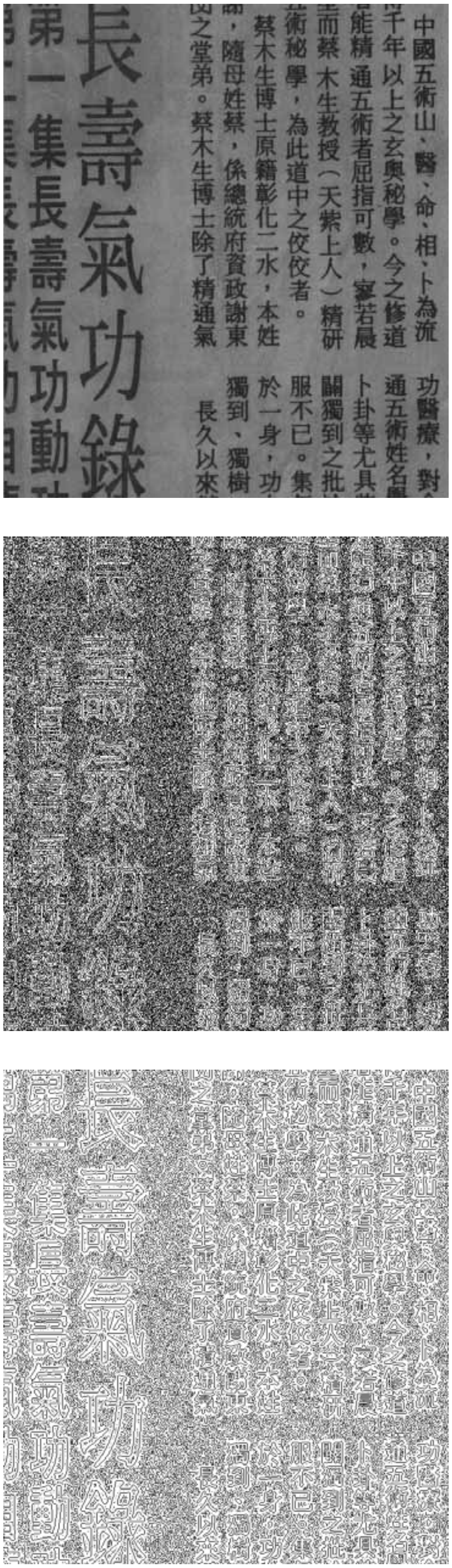


Figure 1: Top: Image contaminated by scanner noise. Middle: Characters extracted from the image. Bottom: Gray scale histogram of the original image.



中國五術山、醫、命、相、卜為流
千年以上之玄奧秘學。今之修道
能精通五術者屈指可數，寥若晨
而蔡木生教授（天紫上人）精研
術秘學，為此道中之佼佼者。
蔡木生博士原籍彰化二水，本姓
，隨母姓蔡，係總統府資政謝東
之堂弟。蔡木生博士除了精通氣

功醫療，對
通五術姓名
卜卦等尤具
闢獨到之批
服不已。集
於一身，功
獨到、獨樹
長久以來

長壽氣功錄

第一集長壽氣功動功

Figure 2: Top: Original image. Middle: Edges at scale 2. Bottom: Edges at scale 4.

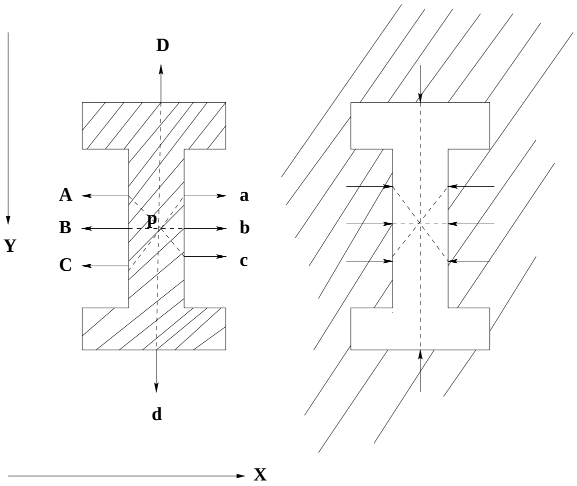


Figure 3: Left: Four pairs of edges: $(A - a)$, $(B - b)$, $(C - c)$, and $(D - d)$ of p, are shown, and the orientations of the edge-pairs are indicated if the character pixels have darker values. Right: The orientations of the edge-pairs if the character pixels have lighter values.

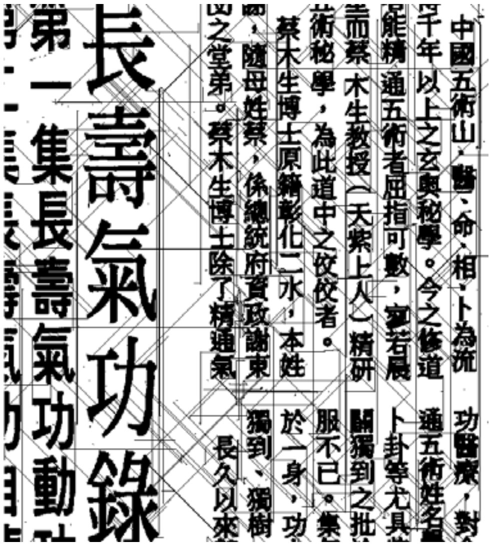
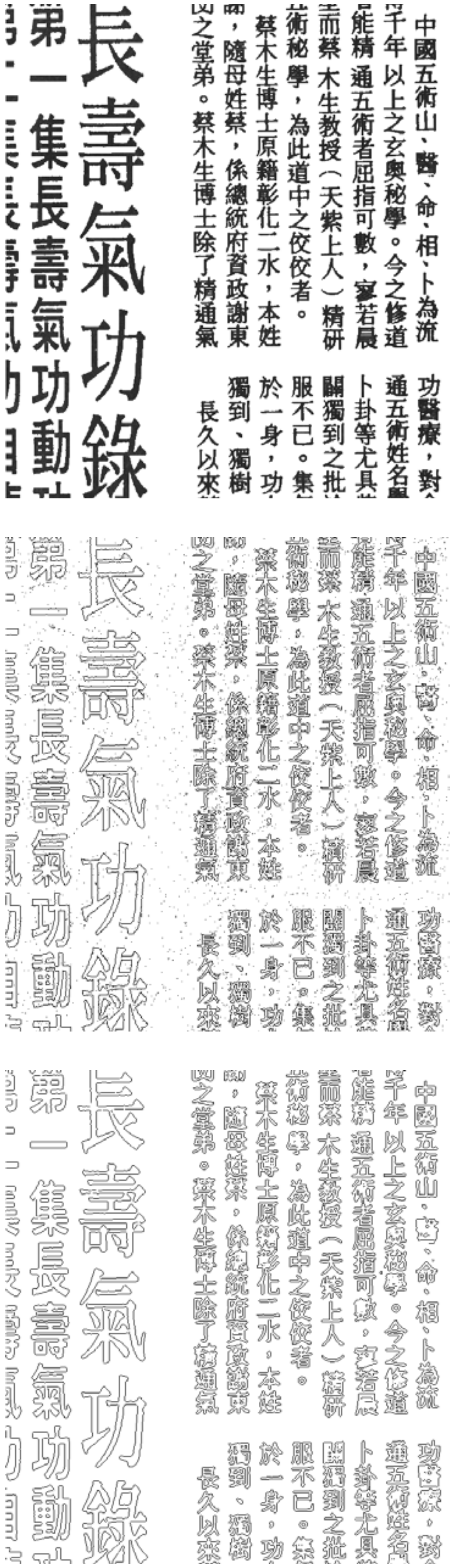


Figure 4: The character map of the original document. Characters grow fat and errors in character edges yield chords between characters.



中國五術山、醫、命、相、卜為流千年以上之玄奧秘學。今之修道能精通五術者屈指可數，寥若晨而蔡 木生教授（天紫上人）精研術秘學，為此道中之佼佼者。

蔡木生博士原籍彰化二水，本姓，隨母姓蔡，係總統府資政謝東之堂弟。蔡木生博士除了精通氣

功醫療，對

通五術姓名

卜卦等尤具

闢獨到之批

服不已。集

於一身，功

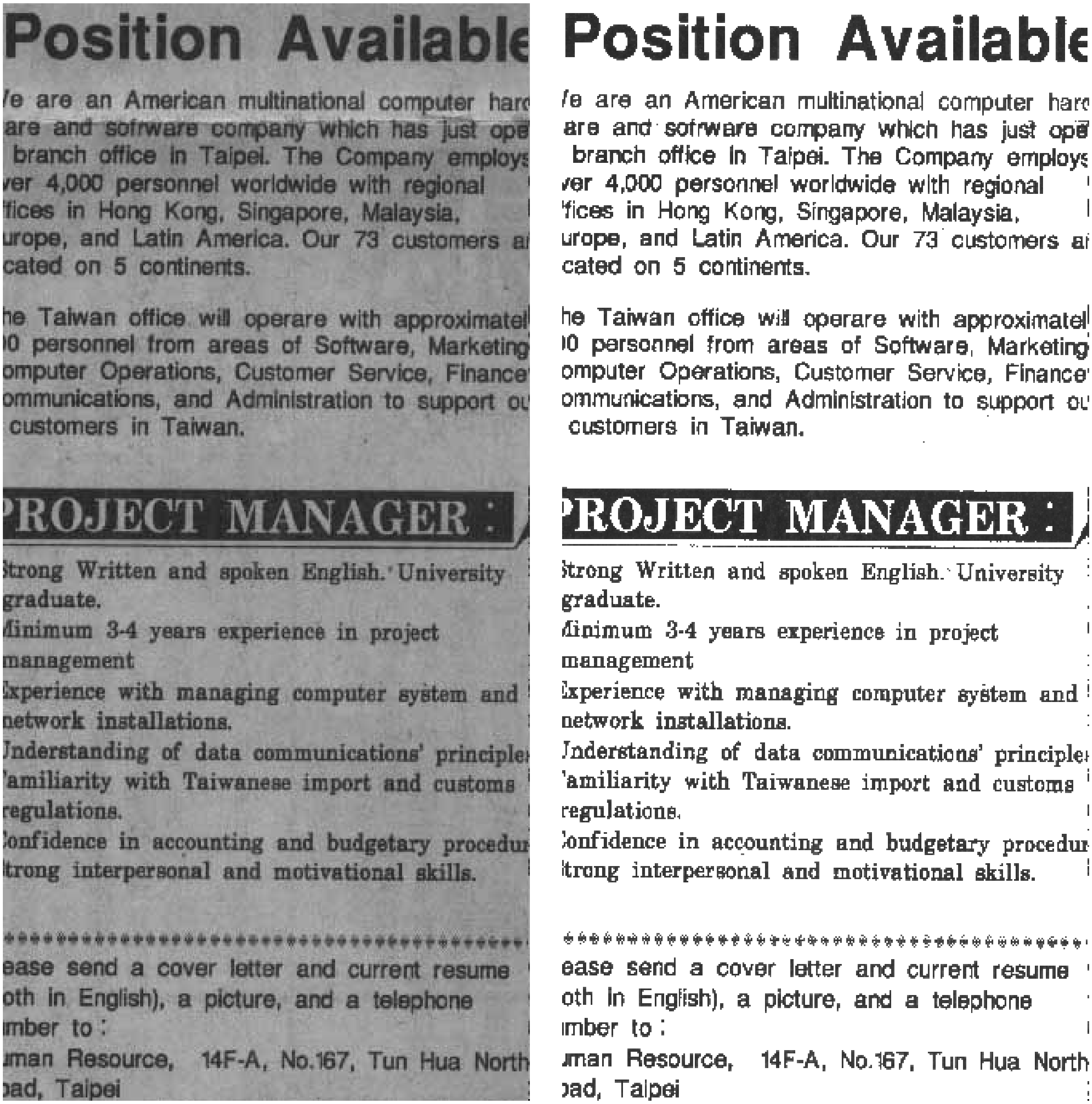
獨到、獨樹

長久以來

長壽氣功錄

第一集長壽氣功動

Figure 5: Top: Character image of the original image. Middle: Contour map at scale 2. Bottom: Contour map at scale 4.



Position Availabl

/e are an American multinational computer harc
are and sofrware company which has just ope
branch office in Taipei. The Company employs
ver 4,000 personnel worldwide with regional
'fices in Hong Kong, Singapore, Malaysia,
urope, and Latin America. Our 73 customers a
cated on 5 continents.

he Taiwan office will operare with approximatel
0 personnel from areas of Software, Marketing
omputer Operations, Customer Service, Finance
ommunications, and Administration to support ou
customers in Taiwan.

ROJECT MANAGER :

Strong Written and spoken English. University
graduate.
Minimum 3-4 years experience in project
management
Experience with managing computer system and
network installations.
Understanding of data communications' principle
'amiliarity with Taiwanese import and customs
regulations.
Confidence in accounting and budgetary procedu
Strong interpersonal and motivational skills.

****

ease send a cover letter and current resume
oth in English), a picture, and a telephone
imber to :
uman Resource, 14F-A, No.167, Tun Hua North
oad, Taipei

Position Availabl

/e are an American multinational computer harc
are and sofrware company which has just ope
branch office in Taipei. The Company employs
ver 4,000 personnel worldwide with regional
'fices in Hong Kong, Singapore, Malaysia,
urope, and Latin America. Our 73 customers a
cated on 5 continents.

he Taiwan office will operare with approximatel
0 personnel from areas of Software, Marketing
omputer Operations, Customer Service, Finance
ommunications, and Administration to support ou
customers in Taiwan.

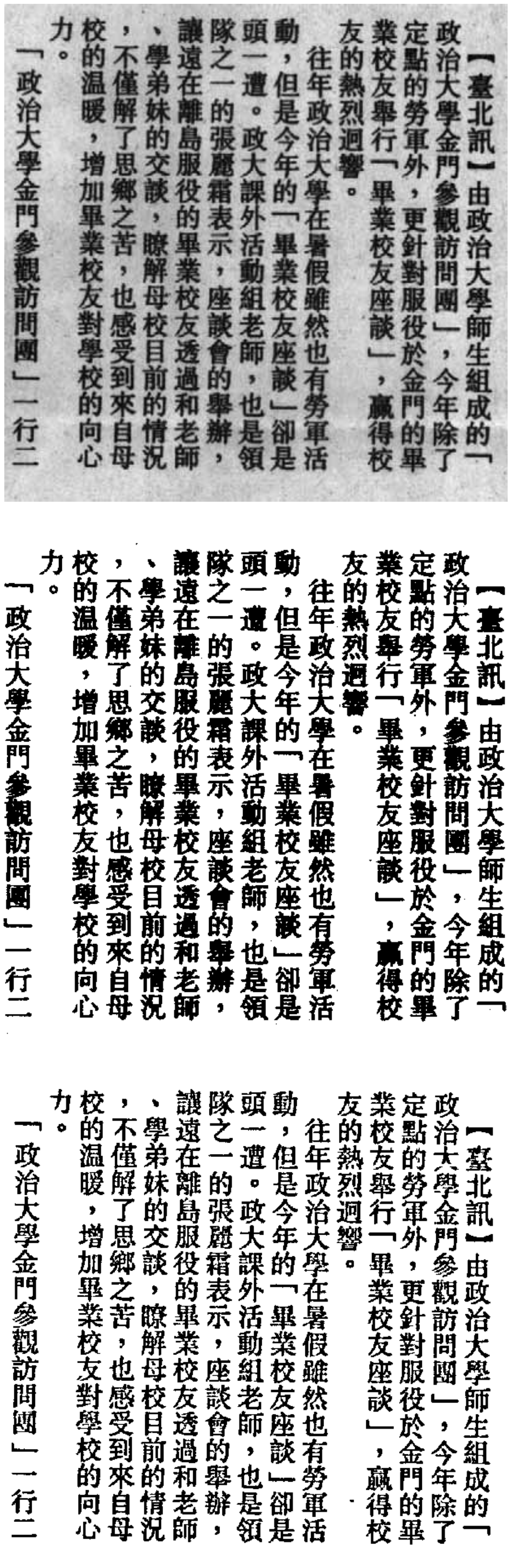
ROJECT MANAGER :

Strong Written and spoken English. University
graduate.
Minimum 3-4 years experience in project
management
Experience with managing computer system and
network installations.
Understanding of data communications' principle
'amiliarity with Taiwanese import and customs
regulations.
Confidence in accounting and budgetary procedu
Strong interpersonal and motivational skills.

****

ease send a cover letter and current resume
oth in English), a picture, and a telephone
imber to :
uman Resource, 14F-A, No.167, Tun Hua North
oad, Taipei

Figure 6: Left: Original document. Right: Character image of the original image.



【臺北訊】由政治大學師生組成的「政治大學金門參觀訪問團」，今年除了定點的勞軍外，更針對服役於金門的畢業校友舉行「畢業校友座談」，贏得校友的熱烈迴響。

往年政治大學在暑假雖然也有勞軍活動，但是今年的「畢業校友座談」卻是頭一遭。政大課外活動組老師，也是領隊之一的張麗霜表示，座談會的舉辦，讓遠在離島服役的畢業校友透過和老師、學弟妹的交談，瞭解母校目前的情況，不僅解了思鄉之苦，也感受到來自母校的溫暖，增加畢業校友對學校的向心力。

「政治大學金門參觀訪問團」一行二

【臺北訊】由政治大學師生組成的「政治大學金門參觀訪問團」，今年除了定點的勞軍外，更針對服役於金門的畢業校友舉行「畢業校友座談」，贏得校友的熱烈迴響。

往年政治大學在暑假雖然也有勞軍活動，但是今年的「畢業校友座談」卻是頭一遭。政大課外活動組老師，也是領隊之一的張麗霜表示，座談會的舉辦，讓遠在離島服役的畢業校友透過和老師、學弟妹的交談，瞭解母校目前的情況，不僅解了思鄉之苦，也感受到來自母校的溫暖，增加畢業校友對學校的向心力。

「政治大學金門參觀訪問團」一行二

【臺北訊】由政治大學師生組成的「政治大學金門參觀訪問團」，今年除了定點的勞軍外，更針對服役於金門的畢業校友舉行「畢業校友座談」，贏得校友的熱烈迴響。

往年政治大學在暑假雖然也有勞軍活動，但是今年的「畢業校友座談」卻是頭一遭。政大課外活動組老師，也是領隊之一的張麗霜表示，座談會的舉辦，讓遠在離島服役的畢業校友透過和老師、學弟妹的交談，瞭解母校目前的情況，不僅解了思鄉之苦，也感受到來自母校的溫暖，增加畢業校友對學校的向心力。

「政治大學金門參觀訪問團」一行二

Figure 7: Top: Original image. Middle: Bilevel image obtained by the scanner. Bottom: Bilevel image obtained using our method. The character recognition rate of the image increased from 75% in the middle to 95% at the bottom.