



PERGAMON

Available at

www.ElsevierComputerScience.com

POWERED BY SCIENCE @ DIRECT®

Pattern Recognition 38 (2005) 289–305

PATTERN  
RECOGNITION

THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

# Extraction of reference lines and items from form document images with complicated background

Dihua Xi, Seong-Whan Lee\*

Department of Computer Science and Engineering, Center for Artificial Vision Research, Korea University, Anam-dong, Seongbuk-ku, Seoul 136-701, Republic of Korea

Received 25 July 2003; accepted 30 April 2004

## Abstract

The extraction of reference lines and items is a fundamental and crucial task in form document analysis. Most of the studies performed so far were done in connection with binary images. This paper proposes a method of extracting lines from gray-level images, by constructing a 2D pseudo Gaussian–Coiflet wavelet with adjustable rectangular support. We also present a method of extracting items using the extracted reference lines and multiresolution wavelet sub-images, which is independent of the intensity of the strokes and backgrounds. The experimental results demonstrate the effectiveness of our proposed methods.

© 2004 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.

*Keywords:* Form document analysis; Reference line extraction; Item extraction; Wavelet; Multiresolution decomposition

## 1. Introduction

A large number of form documents are processed in places such as banks, schools, businesses, and government organizations for many different purposes, such as applications, information collection, subscribing to or canceling a service, etc. A form document usually contains several important items corresponding to important information to be communicated to the addressee. The main purpose of a form document system is to extract and understand these important items. However, it takes much time and effort to process these documents manually. The development of an automatic system for the extraction and recognition of these items of interest is becoming increasingly important in document analysis and understanding.

There are two fundamental and interrelated problems which are encountered when dealing with a typical automation form processing system, the first is *item location and extraction* and the second is *item recognition and interpretation*. The latter concern can be solved by adopting conventional *Optical Character Recognition* (OCR) techniques. In general, most form document systems concentrate only on the former problem.

However, a form document consists of *reference lines* (also called *staff lines* or *baselines*) which are oriented mostly in the horizontal and vertical directions. Once the reference lines have been correctly located, the location of a particular item and the information it contains can be extracted based on some related reference lines. The relation between the items and the reference lines can be described either with *form description language* (FDL) or more simply by means of bounding rectangles. Therefore, the extraction of the reference lines plays a pivotal and essential role in the building of automatic form document systems.

\* Corresponding author. Tel.: +82-2-3290-3197; fax: +82-2-926-2168.

E-mail addresses: dhxi@image.korea.ac.kr (D. Xi), swlee@image.korea.ac.kr (S.-W. Lee).

To this end, over the past few years, numerous form document processing systems have been presented. Most current systems concentrate on the extraction of the fill-in items from a form document [2–6]. Other researchers have concentrated their efforts on one particular sub problem, such as the overlapping problem [7,8] or the relation between the items and the reference lines [9–12]. However, almost all of the current algorithms are based on the extraction of these reference lines. One exception to this rule is a system that was developed by Fan et al. [4], who used the cluster-based technique for the extraction of the characters from the form documents. Fan et al. [4] supposed that all of the strokes belonging to a particular item were close to each other, but distant from other information such as text, reference lines, noise, etc. Therefore, it is impossible to apply this system to forms in which the fill-in items are close to or even overlap the reference lines. Unfortunately, however, most form documents fall into this category. In summary, the extraction of the reference lines is a task that cannot be bypassed, if we are to develop an efficient form processing system.

Form processing systems can be classified into three main types, depending on the type of input form image which is employed. The majority of the current systems [2–7,9–12] are designed to process monochrome document images, in which the forms are required to be converted to binary images. This is the *binary based method*. Another type, which is referred to as the *color based method*, uses color dropout as a means of separating the preprinted entities from the fill-in items [13]. Some of the above binary based systems are effective for forms with simple, fixed backgrounds. In real life, however, forms are becoming more and more complicated with varying styles and complicated backgrounds. The major problem associated with these complicated forms is the poor quality of binarization, due to there being much residual noise, which results in some of the reference lines being absent in the binary image. The main difficulty associated with the color based method is that it is too expensive in terms of both computing power and storage space. Therefore, very little research has been done on the practical application of the color-based method. The third type is the *gray based method* for gray-form images [8,1,14]. Generally, gray-form images contain sufficient information for the distinction to be made between the background and the foreground. Therefore, this method can be use even in the case of complicated forms. Furthermore, it is much less expensive in terms of computer processing and storage space than the color based method. Consequently, in this paper, we focus our attention on the gray based method.

Many algorithms have been developed for the purpose of line detection and extraction. Some approaches, such as strip projection [15] and connected component analysis, are only appropriate for binary images. There are several gray based methods. Ye et al. [16] proposed a simple approach to the extraction of reference lines from gray bank checks, but their method requires that the pixels on the reference lines have a fixed color (usually black). The Hough transform (HT) is

one of the most popular and basic methods for straight-line detection. *Real-time imaging* published a special issue on the HT in April 2000. Briefly, the HT converts each point in Cartesian  $x$ - $y$  image space into Hough  $r$ - $\theta$  space using the parametric representation:  $r = x \cos \theta + y \sin \theta$ . If the points are co-linear, the sinusoids intersect at a point  $(r, \theta)$  corresponding to a parameter of the line. Therefore, the use of the HT is restricted to alignment detection, i.e., the HT can detect only the polar parameters of straight lines, but not the exact position of their pixels. There is currently no algorithm which permits the exact determination of the line pixels using only HT space. Although the theory of the HT sounds simple, the computational complexity involved renders its utilization very expensive in terms of processing time. Tang et al. [1] presented a novel algorithm based on the orthogonal 2D Daubechies wavelet. The computing cost of this method is much lower than that of the HT transform, and the experimental results show that it is effective when the width of the reference lines is appropriate. Unfortunately, however, their method has some disadvantages which are associated with the nature of the orthogonal 2D Daubechies wavelet. Details about the wavelet will be presented in Section 2. Briefly, the wavelet decomposition can only produce a limited number of sub-images, which show the different frequency information of the original image. These sub-images do not show the location of the reference lines, but only contain strong responses around the reference lines. As in the case of the HT method, the wavelet itself cannot extract the reference lines from the form document images. No description of the line extraction algorithm was presented in Ref. [1].

In this paper, we present a generic line extraction algorithm for complicated form documents. A pseudo wavelet function named the *Gaussian-Coiflet (G-C) pseudo wavelet* (referred to below simply as the G-C wavelet) was constructed for the purpose of line extraction. Compared to the 2D Daubechies wavelet with a square support, the G-C wavelet is more suited to the task of line extraction, because it has an adjustable rectangular support. An algorithm based on the strip growth method was developed to extract the reference lines from the series of sub-images produced by the multiresolution wavelet decomposition (MWD) method (Ref. [1] does not use the MWD method). Once the reference lines have been extracted, we also offer a method of performing item extraction using the wavelet sub-images. This method attempts to the fill-in item in a field bounded by some reference lines, and is independent of the color or intensity of the characters, but only requires that the color or intensity of the strokes be different from that of the background.

This paper is organized as follows. Section 2 briefly describes the concept of wavelets and the construction of the pseudo wavelet with an adjustable rectangular support. Details of the proposed method for form document processing are described in Section 3. Our performance evaluation method and experimental results are given in Section 4. Finally, some conclusions are given in Section 5.

## 2. Construction of 2D transforms for reference line extraction

The wavelet theory has become a well-established tool in the field of image processing. In particular, multiresolution analysis has been successfully used in many applications such as image compression, texture analysis and pattern recognition. The application of wavelet transforms to form document analysis was proposed in Ref. [1] by Tang et al. In this section, the prevailing method for the construction of 2D wavelets is introduced. Then, we construct a category of transforms (the 2D pseudo wavelet) with adjustable rectangular supports. Finally, we present the multiresolution decomposition of form documents using the constructed multiresolution transform and show why we chose the Gaussian–Coiflet (G–S) wavelet for the extraction of the reference lines.

### 2.1. 2D Orthogonal wavelet with square support

In the case of one-dimensional space, the continuous wavelet transform of any function  $f(x) \in L^2(R)$  can be defined as a function of two variables

$$W_f(u, s) = \int_R f(x) \frac{1}{\sqrt{s}} \psi^* \left( \frac{x-u}{s} \right) dx$$

which is implemented using a wavelet function  $\psi(x)$ . The function  $\psi(x) \in L^2(R)$  is called a *wavelet function (mother wavelet)* if it satisfies the *admissibility condition*,

$$\int_R \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < +\infty,$$

where  $\hat{\psi}(\omega)$  is the Fourier transformation of  $\psi(x)$ . The admissibility condition also implies that  $\int_R \psi(x) dx = \hat{\psi}(0) = 0$ , which guarantees that the wavelet is well localized. In practice, wavelet functions are usually normalized to have unit energy, i.e.  $\|\psi(x)\|^2 = \int_R |\psi(x)|^2 dx = 1$ .

One of the most important challenges facing scientists working in this domain is how to construct a proper wavelet, especially one which reflects the properties of a real application. A number of wavelet functions have been constructed by many researchers. However, Mallat [17] presented a theoretical frame of multiresolution analysis which constitutes the fundamental concept necessary to construct and understand the wavelet paradigm.

Mallat [17] first introduced the multiresolution approximation, which is a sequence of nested closed subspaces,  $V_n \in L^2(R)$ ,  $V_n \subset V_{n+1}$ ,  $n \in Z$ , which satisfies several limitations. A scaling function  $\varphi(x) \in V_0$  is then defined whose integer-translates span the space  $V_0$ . It is required that the scaling function satisfy  $\int_R \varphi(x) dx \neq 0$  and

$$\varphi(x) = \sqrt{2} \sum_{k \in Z} h_k \varphi(2x - k) \tag{1}$$

for some coefficients  $h_k$ . The wavelet function  $\psi(x)$  is finally constructed using the scaling function  $\varphi(x)$ . More details concerning the construction of wavelet function based on multiresolution analysis theory can be found in most text books on the subject of wavelets [18].

Let us suppose that  $\psi(x)$  and  $\varphi(x)$  are the wavelet and the scaling functions of a 1D wavelet, respectively. The traditional construction of a 2D wavelet can be implemented from an 1D wavelet by

$$\begin{aligned} \Phi(x, y) &= \varphi(x)\varphi(y), & \Psi^1(x, y) &= \varphi(x)\psi(y), \\ \Psi^2(x, y) &= \psi(x)\varphi(y), & \Psi^3(x, y) &= \psi(x)\psi(y), \end{aligned} \tag{2}$$

where  $\Phi(x, y)$  is the scaling function and  $\Psi^i(x, y)$ ,  $i = 1, 2, 3$  are wavelet functions.

In practice, we usually use the compact wavelet whose support is in a finite interval (being zero out of the support). Since the sizes of the supports for both the scaling and wavelet functions of a 1D compact wavelet are always equal, the support of a 2D compact wavelet, which is constructed as defined in Eq. (2), is usually a square area.

Tang et al. [1] proposed a new method of extracting reference lines from documents with gray level backgrounds. They used a 2D wavelet which is constructed using Eq. (2) by means of the 1D Daubechies wavelet [19] with 5 vanishing moments. The wavelet function  $\varphi(x)$  and scaling function  $\psi(x)$  are illustrated in Fig. 1.

In this approach, however, the authors presented a method of processing form documents with gray backgrounds. The experimental results showed that Tang’s method constituted promising technique for extracting reference lines.

Unfortunately, some difficulties are encountered when attempting to implement the method presented in Ref. [1]. Fig. 2 shows the simplest case, which illustrates the problems involved, by showing the responses to horizontal lines with different widths in the LH sub-images. These problems can be summarized as

- *Response pervasion.* The response of a line is spread out over a band shaped area, which extends 10 pixels to each side of the line edges in the original image. This problem makes it difficult to obtain the exact location of the line, and therefore affects the item extraction as well as the item recognition processes, especially when the item is close to the line. It only makes matters worse that one or two peaks may appear near the edge of the line.
- *Shift sensitivity.* The responses to horizontal (vertical) lines vary considerably when the line is shifted one-pixel in the vertical (horizontal) direction. This problem can be visualized in Fig. 2 by observing the visible difference between (b) and (c), which are the responses to (a) and (d). In Fig. 2, (d) represents (a) shifted vertically by one-pixel. Fortunately, this is a systematic displacement of 2 pixels caused by the discrete dyadic wavelet decomposition.

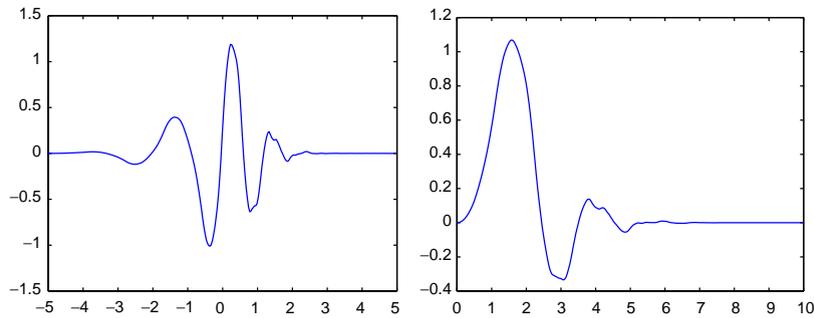


Fig. 1. Daubechies wavelet and scaling functions with 5 vanishing moments.

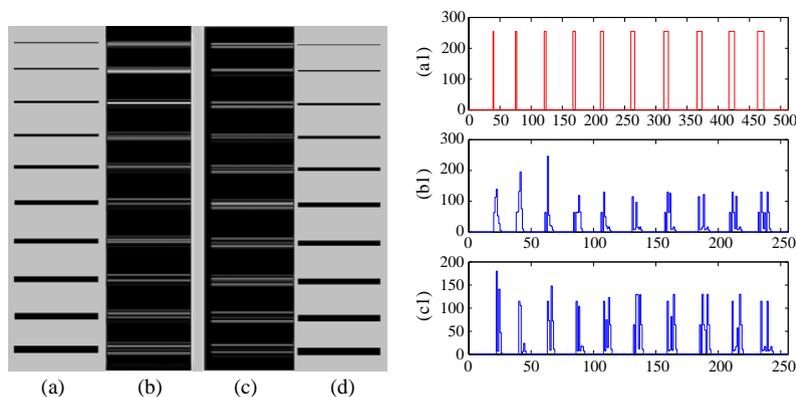


Fig. 2. Responses of horizontal lines to 2D Daubechies wavelet used in Ref. [1]. (a) Horizontal lines with widths varying from 1 to 10 pixels; (b) LH wavelet sub image of (a); (c) LH sub-images of (d); (d) image shifted by one pixel in the vertical direction. (a1), (b1) and (c1) are the gray values for one vertical line in (a), (b) and (c). To compare the responses with those of the original lines, images (a) and (d) are displayed at half the actual size.

- *Width sensitivity.* The result is sensitive to variations in the width of the line. Better results were obtained when the line width was 2 or 3 pixels than in all other cases.

The above problems only concern the simplest case (black lines on a white background). In fact, the problems may become worse when the issue of noise (such as that originating from text, background pictures, etc.) is included in the case of more complicated images (for example, the image in Fig. 6). The main reason for the above problems is the use of a discrete wavelet with a square support. If we were choose a wavelet with a smaller support, then the responses would be more centralized with respect to the line and the above problems would be alleviated. However, if the support square is too small, it will be impossible to obtain a strong enough response near the line and this will make it difficult to distinguish the line from other information, such as text characters or strokes. In summary, there will always be some drawbacks when using the square support wavelet, irrespective of whether a small or large support is selected.

## 2.2. Construction of 2D transforms with adjustable rectangular supports

From the above analysis, it can be seen that a type of wavelet needs to be found which can collect enough information about the line and whose response is centralized with respect to the line. A wavelet with an adjustable rectangular support may meet this requirement.

Indeed, the 2D Daubechies wavelet used in Ref. [1] is an orthogonal wavelet. The use of this kind of wavelet is absolutely necessary when the reconstruction of the image is required, for example in image compression. However, wavelet orthogonality is not so important for pattern feature extraction. In the Section 3, we will define 2D transforms with an adjustable rectangular support. Although, strictly speaking, these transforms do not fit into the traditional definition of wavelets, and therefore cannot be used to reconstruct the original signal, we nevertheless refer to them as *2D wavelets* (they are also sometimes referred to as *pseudo wavelets*). Using these constructed wavelets, however, a

better result can be obtained in the case of line extraction. It is known that the 1D wavelet function is defined by the admissibility condition  $\int_{\mathbb{R}} (|\hat{\Psi}(\omega)|^2/|\omega|) d\omega < \infty$ . Similarly, we can define the 2D wavelet functions using the 2D admissibility condition.

**Definition 2.1.** Suppose  $\Psi(x, y) \in L_2(\mathbb{R}^2)$ , then  $\Psi(x, y)$  is a 2D (pseudo) wavelet function if and only if it satisfies the admissibility condition

$$\int \int_{\mathbb{R}^2} \frac{|\hat{\Psi}_i(\omega_x, \omega_y)|^2}{\sqrt{\omega_x^2 + \omega_y^2}} d\omega_x d\omega_y < +\infty \tag{3}$$

where  $\hat{\Psi}_i(\omega_x, \omega_y)$  is the two-dimensional Fourier transformation of  $\Psi(x, y)$ .

Using the above definition of the two dimensional wavelet function, we can construct a series of transformations (scaling and wavelet functions) by means of the following theorem.

**Theorem 2.1.** Let  $\psi(x)$  and  $\phi(x)$  be the wavelet and scaling functions of a 1D wavelet, respectively, and  $g(x) \in L^2(\mathbb{R})$ , i.e.

$$\|g(x)\| = \int_{\mathbb{R}} |g(x)|^2 dx < +\infty.$$

Then (i)  $\Psi_1(x, y) = \psi(x)g(y)$  and  $\Psi_2(x, y) = g(x)\psi(y)$  are 2D wavelet functions; (ii)  $\Phi(x, y) = \phi(x)\phi(y)$  is a 2D scaling function.

$\Phi(x, y)$  is a scaling function because its construction is the same as the traditional construction of a 2D wavelet. To prove that  $\Psi_i(x, y)$  is a wavelet function, we only need to prove that each wavelet function satisfies the admissibility condition (3).

**Proof.** We only give the proof of  $\Psi_1(x, y) = \psi(x)g(y)$ . Since  $\Psi_1(x, y)$  is a separable function, by referring to the property of the 2D Fourier transforms, we have

$$\hat{\Psi}_1(\omega_x, \omega_y) = \hat{\psi}(\omega_x)\hat{g}(\omega_y).$$

Therefore,

$$\begin{aligned} & \int \int_{\mathbb{R}^2} \frac{|\hat{\Psi}_1(\omega_x, \omega_y)|^2}{\sqrt{\omega_x^2 + \omega_y^2}} d\omega_x d\omega_y \\ &= \int \int_{\mathbb{R}^2} \frac{|\hat{\psi}(\omega_x)|^2 |\hat{g}(\omega_y)|^2}{\sqrt{\omega_x^2 + \omega_y^2}} d\omega_x d\omega_y \\ &\leq \int_{\mathbb{R}} \frac{|\hat{\psi}(\omega_x)|^2}{|\omega_x|} d\omega_x \int_{\mathbb{R}} |\hat{g}(\omega_y)|^2 d\omega_y < +\infty. \end{aligned} \tag{4}$$

**Remarks.**

- The admissibility condition implies that

$$0 = \hat{\Psi}_i(0, 0) = \int \int_{\mathbb{R}^2} \Psi_i(x, y) dx dy. \tag{5}$$

- If  $g(x)$  is subject to  $\|g(x)\|^2 = 1$ , then  $\Psi_i(x, y)$  also has unity energy

$$\begin{aligned} \|\Psi_1(x, y)\|^2 &= \int \int_{\mathbb{R}^2} |\Psi_1(x, y)|^2 dx dy \\ &= \int \int_{\mathbb{R}^2} |\psi(x)|^2 |g(y)|^2 dx dy \\ &= \int_{\mathbb{R}} |\psi(x)|^2 dx \int_{\mathbb{R}} |g(y)|^2 dy = 1. \end{aligned} \tag{6}$$

In general, the wavelet defined in Theorem 2.1 is not orthogonal, due to the arbitrariness of  $g(x)$ , even if the wavelet defined by  $\phi(x)$  and  $\psi(x)$  is orthogonal. Indeed, according to this theorem, any function  $g(x)$  with non-zero finite energy can be used to construct a 2D wavelet, because  $g(x)$  can be normalized by  $g(x)/\int g(x) dx$ . For example, the 2D orthogonal Daubechies Wavelet is a special case of the theory obtained by setting the 1D Daubechies scaling function  $\phi(x)$  and the wavelet function  $\psi(x)$  to be  $g(x)$  in Theorem 2.1.

If we choose the compact 1D wavelet, the support of the 2D scaling function  $\Phi(x, y)$  is still a square, as in the case of the traditional wavelet. In fact,  $\Phi(x, y)$  is no use, if the multiresolution decomposition is not included in a system such as [1]. Using the above theorem, the supports for the wavelet functions  $\Psi^1$  and  $\Psi^2$  can be defined as adjustable rectangular supports which depend on the support of  $g(x)$ . The construction can easily be implemented by choosing different supports for  $\psi(x)$  and  $g(x)$  according to the above theorem. The supports used for the decomposition of a given function,  $f(x, y)$ , using the traditional wavelet (right) and the previous constructed wavelet (left) are shown in Fig. 3. From this comparison, for the traditional wavelet, not enough information is collected if the support is too small (upper-right in Fig. 3), while the computing cost is too high and the wavelet is very noise-sensitive in the case of a large support (bottom-right in Fig. 3). Our method can overcome these drawbacks effectively.

Furthermore, using the constructed 2D wavelet defined in Theorem 2.1, any function  $f(x, y)$  can be decomposed into 3 sub functions which indicate different frequency information. The related 2D continuous wavelet decomposition can be estimated by

$$\begin{aligned} f_{LH}(x, y) &= \frac{1}{2} \int \int_{\mathbb{R}^2} f(u, v) g\left(\frac{x-u}{2}\right) \psi\left(\frac{y-v}{2}\right) du dv, \end{aligned} \tag{7}$$

$$\begin{aligned} f_{HL}(x, y) &= \frac{1}{2} \int \int_{\mathbb{R}^2} f(u, v) \psi\left(\frac{x-u}{2}\right) g\left(\frac{y-v}{2}\right) du dv \end{aligned} \tag{8}$$

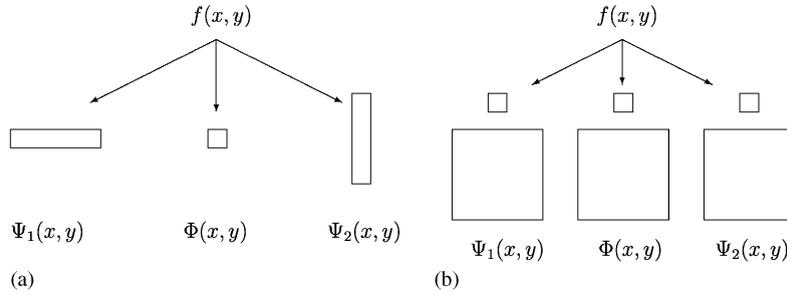


Fig. 3. Support comparison of the wavelet functions (left and right) and scaling function (center) in (a) the constructed functions and (b) the traditional wavelet.

which show the low–high and high–low frequency information of  $f(x, y)$  in the horizontal–vertical order directions. Moreover, the transformation of scaling function

$$f_{LL}(x, y) = \frac{1}{2} \int \int_{R^2} f(u, v) \varphi\left(\frac{x-u}{2}\right) \varphi\left(\frac{y-v}{2}\right) du dv \quad (9)$$

shows the low–low components of  $f(x, y)$ .

If we extend this operation to the low–low component,  $f_{LL}(x, y)$ , multiresolution wavelet decomposition can be implemented as shown in Fig. 4 and a series of sub-images are produced.

The sub-images  $f_{LH}^i(x, y)$  and  $f_{HL}^i(x, y)$  are estimated from  $f_{LL}^{i-1}(x, y)$  by means of the wavelet functions  $\Psi^1$  and  $\Psi^2$ , respectively. If we suppose that  $f_{LL}^0(x, y) = f(x, y)$ , the sub-image  $f_{LL}^i(x, y)$  which is used for the next decomposition is produced from the previous level  $f_{LL}^{i-1}(x, y)$  by the scaling functions  $\Phi$ . Therefore, a series of sub functions  $\{f_{LH}^i(x, y), f_{HL}^i(x, y)\}$  ( $i = 1, \dots, N$ ) can be produced using this multiresolution wavelet decomposition. The sub-image sequence of  $f_{HL}^i(x, y)$  contains the multi scales information of the horizontal lines, while  $f_{LH}^i(x, y)$  contains the multi scales information of the vertical lines. We will use these sub images for the purpose of line extraction in Section 3.

So far, we have compared various kinds of 2D wavelets constructed by several known wavelet functions, including Daubechies orthogonal wavelets, Coiflets, Symlets, and some other biorthogonal wavelets. We reached the conclusion that the most accurate experimental results can be obtained through the 2D wavelet which is constructed using the Coiflet wavelet and Gaussian function, i.e.

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/2\sigma^2}.$$

This type of wavelet is called a 2D Gaussian–Coiflet(G–C) wavelet. The G–C wavelet function  $g(x)\psi(y)$  is illustrated in Fig. 5a, for which the corresponding 1D wavelet function

$\psi(x)$  and the Gaussian function  $g(x)$  are illustrated in Fig. 5b & c, respectively. In fact, the Gaussian function is not a compact function. We limit the support for the Gaussian function in  $[-3\sigma, 3\sigma]$ . According to the  $3\sigma$  rules of the Gaussian function, the discrete signal after discretization is almost no different between using the  $3\sigma$  limitation of support and using the non-limited function.

The Coiflet wavelet originates from the Daubechies wavelet, but is more symmetric. It is referred to as the Coiflet wavelet because it was constructed by Daubechies [19] at the request of Coifman. Therefore, although the Coiflet wavelet and the well-known Daubechies wavelet are similar at a certain level, the Coiflet wavelet does have some differences, in that it was constructed with vanishing moments, not only for wavelet function  $\psi(x)$ , but also for scaling function  $\varphi(x)$ . In the Coiflet wavelet formulation process, the following equations must be satisfied:

$$\int_R x^l \psi(x) dx = 0, \quad \int_R \varphi(x) dx = 0, \\ \int_x^l \varphi(x) dx = 0, \quad (l = 0, 1, \dots, L - 1), \quad (10)$$

where  $L$  is the order of the Coiflet wavelet function, and  $\varphi(x)$  is a scaling function associated with  $\psi(x)$ . The near symmetric property of this wavelet is desirable in the case of line extraction, due to the strong response at the center of the line. The Gaussian function is not sensitive to the high frequency in a signal, but is nevertheless able to collect the information. It is easy to adjust the support of the Gaussian function by varying the parameter  $\sigma$ .

Compared to the previous wavelet based method [1], the problems of response pervasion and shift sensitivity can be alleviated by using the G–C wavelet. In our system, the problem of width sensitivity was alleviated by using the principle of multiresolution decomposition, which will be described in Section 3.

Compared to many other approaches, another consequential advantage of the G–C wavelet based method is that it works very well when the original image contains much

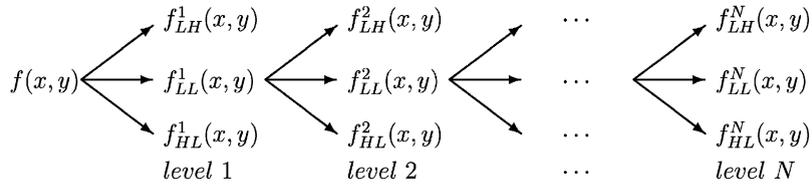


Fig. 4. Multiresolution wavelet decomposition using the constructed wavelet with rectangular support.

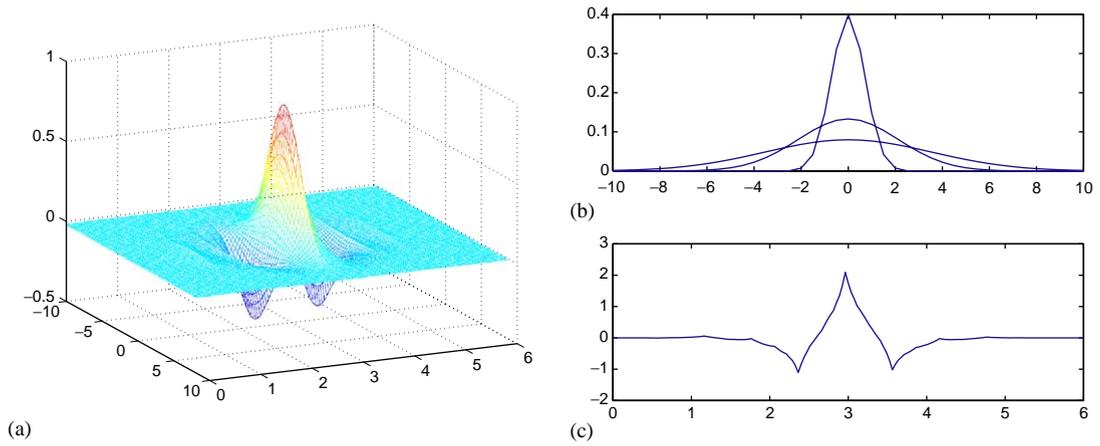


Fig. 5. Construction of Gaussian–Coiflet (G–C) wavelet. (a) 2D G–C wavelet constructed by: (b) 1D Gaussian function with  $\sigma = 1, 3, 5$ ; and (c) Coiflet wavelet with vanishing moment 1.

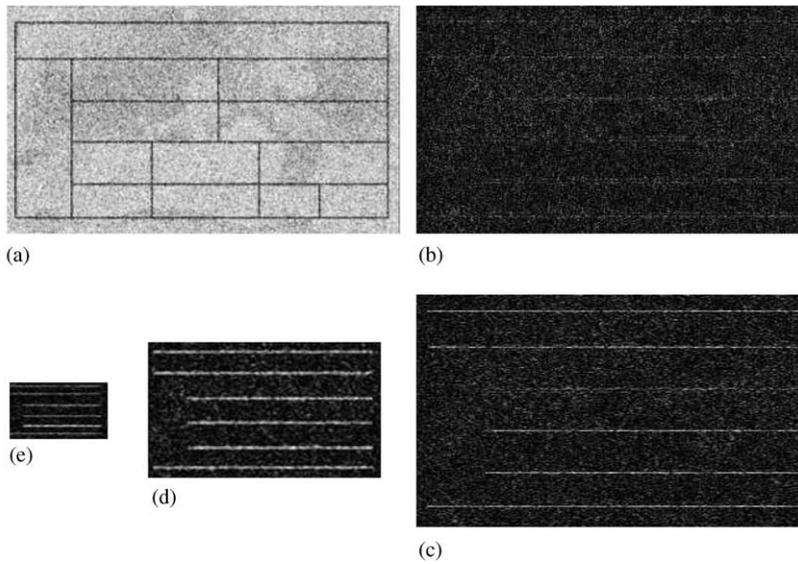


Fig. 6. Responses to a form with heavy noise. (a) Input image; (b) LH sub-images generated by means of the wavelets used in Tang's paper [1]; and multiresolution wavelet decomposition by G–C at: (c) level 1; (d) level 2; (e) level 3.

noise. It is impossible to use a projection based method to process an image with heavy noise, as can be seen in the example shown in Fig. 6a. Tang's method [1] may not be very sensitive to noise, but our G–C wavelet based method can produce a better result than the method described in Ref. [1]. Fig. 6 shows an example of the processing of a form with heavy noise. Fig. 6 (c–e) illustrates the sub-images at levels 1–3 which were treated by the G–C multiresolution wavelet. Our method produces much stronger responses around the lines (Fig. 6b) than the method described in Ref. [1] (Fig. 6c) and it is possible to extract the lines from coarse to fine, i.e. by first using the smallest sub image and then verifying the extracted lines using the larger sub-images at higher level(s). Therefore, our method can be applied to form documents with a very complicated background, and can save computing cost by processing from coarse to fine.

### 3. Form document processing

In this section, the processing of an input form document will be illustrated step-by-step. The multiresolution wavelet decomposition method and its implementation by the 2 times 1D filters are first studied. Then, we propose the algorithm to be used for the extraction of the horizontal and vertical lines, as well as that of whole form table, by the strip growth method, using the sequence of sub-images. Finally, the items of interest can be located and extracted based on the extracted reference lines.

#### 3.1. Estimation of 2D wavelet decomposition

Using the multiresolution wavelet decomposition method shown in Fig. 4, an input image  $f(m, n)$  can be decomposed into a series of LH and HL sub-images by means of the 2D pseudo wavelet proposed in Theorem 2.1. Please note that the wavelet and scaling functions are separable, and therefore the integrals in Eqs. (7)–(9) can be converted into repeated integrals in 1D. In the discrete case, therefore, suppose that  $L_g$  is a filter for the Gaussian function  $g(\cdot)$ , and  $H$  and  $L$  are filters related to the wavelet function  $\psi(\cdot)$  and the scaling function  $\varphi(\cdot)$ , respectively. The 2D G–C wavelet decomposition for estimating each sub image of a discrete image  $f(m, n)$  can be estimated as two repeated 1D filters as illustrated in Fig. 7.

For a given filter  $\mathcal{K} = \{k_1, k_2, \dots, k_N\}$  with support  $N$ , the filtered results along the horizontal or vertical direction for an image  $f(m, n)$  can be produced by

$$f^x(m, n) = \sum_{i=1}^N k_i f(2 * m + i, n), \quad (11)$$

$$f^y(m, n) = \sum_{i=1}^N k_i f(m, 2 * n + i), \quad (12)$$

where  $f^x$  and  $f^y$  represent the filtered results obtained with filter  $\mathcal{K}$  (named  $\mathcal{K}^x$  and  $\mathcal{K}^y$ ) in the horizontal and vertical directions, respectively. Fig. 8 provides an example of the multiresolution wavelet decomposition of a form document image. This is a very complicated form which was scanned under poor lighting conditions and the signal of some lines is very weak. We will use this example to illustrate our algorithm. To save space, we do not show the original image, which is similar to the LL sub-images but is at twice the size of the LL sub image at level 1 (the bottom right image shown in Fig. 8).

#### 3.2. Horizontal and vertical line extraction

A series of LH and HL sub-images of an input form image were produced using the multiresolution G–C wavelet decomposition method, as described in the previous sections. No matter how much difference in color or intensity there is between the lines and the background, strong responses can be obtained with respect to the horizontal or vertical lines in the series of LH or HL sub-images, only in so far as a strong contradistinction exists between the line and the near background. Unfortunately, strong responses are also obtained from some other components such as the printed texts, although most of them can be restricted. In this section, we propose a method of preserving the lines and removing the noise from the series of LH and HL sub-images. The method of coarse to fine is used for line searching, i.e. the long line is roughly located in the small sub-images and then the extracted lines are verified in the larger sub-images. The steps required to extract the form lines are as follows.

*Step 1:* Definition of the shortest line lengths,  $L_{min}^H$  and  $L_{min}^V$ , which should be shorter than the shortest horizontal and vertical reference lines to be estimated, respectively. Nevertheless, if the length of the shortest line is too large, then  $L_{min}^H$  and the  $L_{min}^V$  should be set much smaller than the shortest reference lines, so that the system can be applied to slightly skewed forms. However, the length of the shortest lines should be large enough so that any hand written strokes or printed texts will be excluded.

*Step 2:* Extraction of the reference lines from the smallest sub image. To limit the time spent searching and alleviate the problem of shift sensitivity, we start searching in the smallest sub-images, whose width and height are only  $\frac{1}{2}^N$  ( $N$  is the max decomposition level and is usually set to 2–5, depending on the complexity of the processing form) of the input image. The main feature of the algorithm used for treating the horizontal lines can be described as follows.

- *Threshold selection.* The responses to most pixels are close to zero in the LH and HL sub-images at all levels. We divide all the pixels in a sub image into three sets by means of two thresholds  $T_1$  and  $T_2$  ( $T_1 < T_2$ ): *weak*

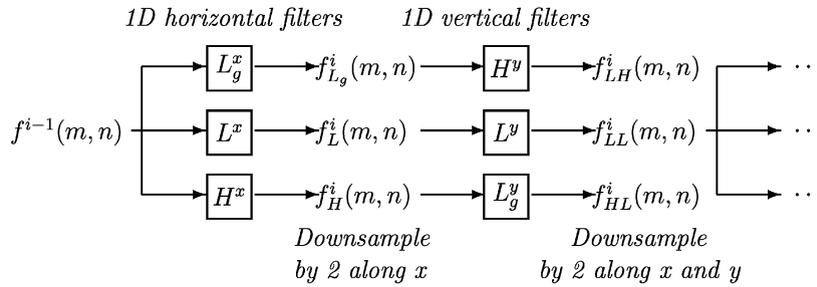


Fig. 7. Estimating the Gaussian–Coiflet wavelet decomposition by means of two 1D filters.

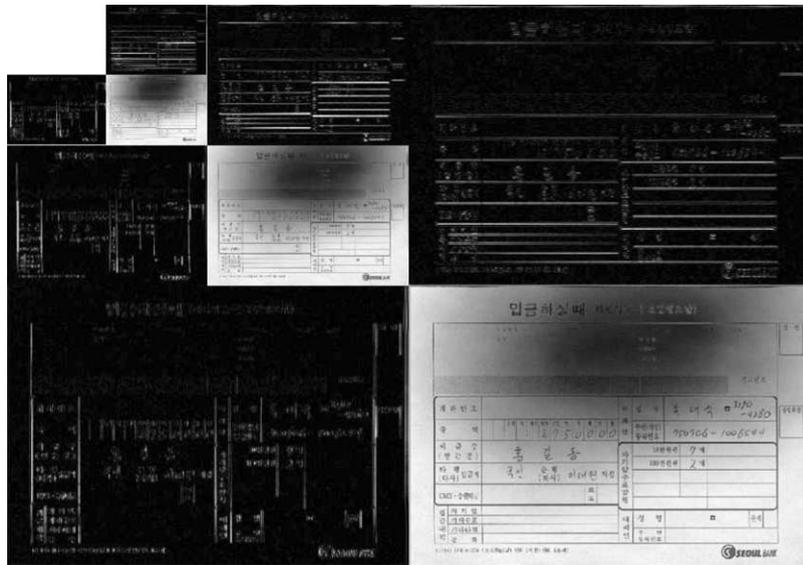


Fig. 8. Multiresolution G–C wavelet decomposition of a form document image. The LH, HL and LL sub-images at levels 3 to 1 shown at the top, left and diagonal, respectively.

response set:  $S_w = \{(x, y) | f_{LH}(x, y) \leq T_1\}$ , normal response set:  $S_n = \{(x, y) | T_1 < f_{LH}(x, y) \leq T_2\}$  and strong response set:  $S_s = \{(x, y) | f_{LH}(x, y) > T_2\}$ . In practice,  $T_1$  and  $T_2$  are usually set to about 10 and 30, respectively. For a form document image which is not too complex,  $S_w$  may contain more than 90% of the pixels, while each of the other two sets,  $S_n$  and  $S_s$ , contains about 5%. A pixel situated on a line should be in set  $S_n$  or  $S_s$ .

- **Line extraction by strip growth.** For a given pixel  $P \in S_n \cup S_s$ , suppose  $S_P$  denotes the following  $L_{min}^H$  ( $L_{min}^V$ ) pixels of P in the LH (HL) sub image. Then,  $S_P$  is not on a line if: (a)  $S_P$  contains some pixels in  $S_w$ ; (b) the pixels in  $S_P$  vary frequently in  $S_n$  and  $S_s$ . This case may arise in an area containing printed texts or strokes. On the other hand,  $S_P$  is a part of one line and P is the start point of the line if: (a)  $S_P \subset S_n$  or

$S_s$  almost everywhere (a.e.) which means that  $S_P$  is a subset of  $S_n$  or  $S_s$  for exception of few pixels, which may be overlapped by other lines or text strokes; (b) the pixels in  $S_P$  vary slowly and interactively in  $S_n$  and  $S_s$ , which may happen in the case of a skewed line or if there is a change in luminosity. If  $S_P$  is a part of a line, the length of the extracted line  $S_P$  is extended so as to extract the whole single horizontal or vertical line (*strip growth for single line extraction*). Then, slightly moving the extracted line in its perpendicular direction enables us to extract the whole line (*strip growth for whole line extraction*). A concise block diagram of this algorithm is shown in Fig. 9.

**Step 3:** Verification of the extracted lines with the help of the ensuing bigger sub-images. The extracted lines in Step 2 are not well situated with respect to the original lines,

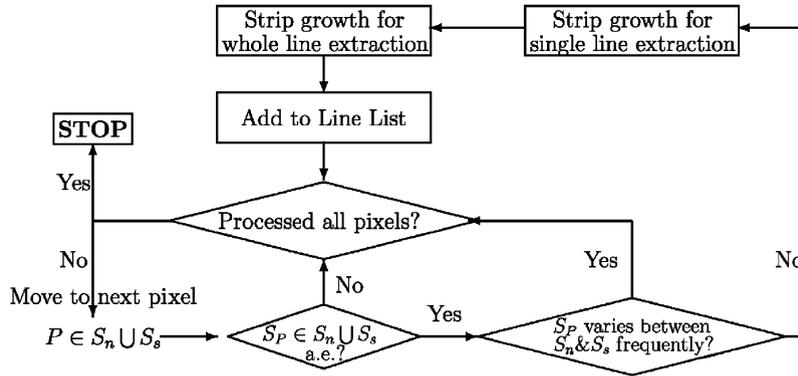


Fig. 9. Block diagram of the proposed algorithm for line extraction from the smallest sub image after categorizing by threshold.  $S_n$  and  $S_s$  denote the sets for the pixels with normal and strong responses in the sub-image, respectively.  $S_P = \{(x_P + i, y_P) | 0 < i < L_{min}^H\}$  for LH sub images or  $S_P = \{(x_P, y_P + i) | 0 < i < L_{min}^V\}$  for HL sub-images, where  $(x_P, y_P)$  is the coordinate of pixel  $P$ . The symbol ‘a.e.’ (almost everywhere) means that this property holds in a set except for a few pixels.

because we only used the smallest sub-images where one pixel corresponds a  $2^N \times 2^N$  block in the original image. It is still necessary to locate the lines precisely by verifying these extracted lines in the following larger sub-images, level by level, ending with level 1. A process similar to that used in Step 2 was applied to the extracted line pixels only.

For the example in Fig. 8, we start the search at level 2, because some horizontal lines at the bottom left are very close to each other. These neighboring lines may be indistinguishable in the sub image, because it is too small. In this example, we set the thresholds at  $T_1 = 10$  and  $T_2 = 35$ . We set the lengths of the horizontal and vertical shortest lines to  $L_{Min}^H = 40$  and  $L_{Min}^V = 30$ , the corresponding values of  $(L_{Min}^H, L_{Min}^V)$  in the sub-images at level 2 and 1 are (10, 8) and (20, 16). The extracted horizontal and vertical lines based on the previous algorithm are combined into a single image and illustrated in Fig. 10(a).

The start and end points of an extracted line may not completely match those of the original line after simply combining the horizontal and vertical lines, as shown in Fig. 10(a). Therefore, we may need to check and adjust the end-points of the extracted lines. We check whether the end point of an extracted line is on another extracted perpendicular line. If not, we check whether the distance from the end point to one of the extracted lines is less than the threshold. If both the “L” and “T” shapes are found for an end point, we choose the shape of the end point as the “L” (corner) shape (a corner) but not choosing the “T” shape. If such a line is found, then the end point is taken to be a redundant start point and is therefore eliminated. Fig. 10(b) illustrated the adjusted form lines of Fig. 10(a).

### 3.3. Item extraction

After extracting all of the reference lines, we can determine the field containing the item of interest by making use of the particular structure of a given form. The identification of a field can be accomplished by using the positional relation of the item to the reference lines. One example of this would be the case where a particular field happens to be situated between some horizontal and vertical reference lines. In the example shown in Fig. 8, the strokes of the customer are located in the rectangle limited by the second and third vertical and the fourth and fifth horizontal lines. Thus, using this information, we can obtain the rectangular region pertaining to this item.

The fill-in strokes are usually accompanied by strong responses in both the LH and HL sub-images, because the intensities of the background and the strokes are likely to be different. Therefore, we can locate and extract the fill-in item in the extracted item rectangle. In practice, the strong response of  $f_{LH}^2(x, y) + f_{HL}^2(x, y)$  from the smallest to the largest sub-images is applied to extract the location of the fill-in strokes information. The algorithm used for the extraction of the item can be summarized by means of the following steps.

- *Rough item location.* Initial search for the strongest responses  $f_{LH}^2(x, y) + f_{HL}^2(x, y)$  in the extracted rectangle at the smallest level  $N$  decided by the rectangular size. The simple smooth filter is used and the region containing the strokes produces much stronger responses than the background. Therefore, it is easy to obtain the threshold required to distinguish the regions containing strokes from the background.

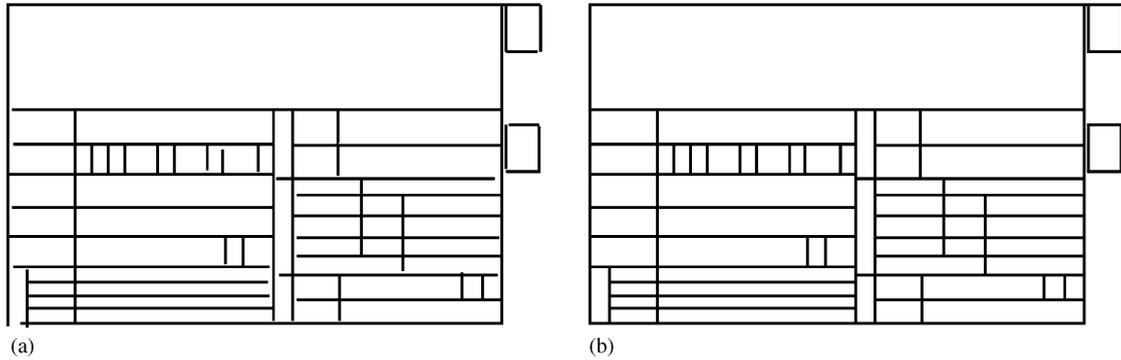


Fig. 10. (a) The extracted horizontal and vertical lines of the example in Fig. 8; (b) the merged form structure of (a).

- *Location verification.* A more exact location of the region containing the strokes can be obtained by means of the ensuing larger sub-images, using an approach similar to that described in the previous step. This verification step is limited to the extracted regions. The resulting location should be better than the initial one, because we use the larger sub-images, which contain finer information pertaining to the image.
- *Fill-in item strokes extraction.* The main background color (gray intensity) may be estimated from the region near to the extracted strokes in the original image. Then, the stroke color can be decided in the extracted region, because the main color is different from that of the background. Furthermore, the strokes can be extracted by making use of the stroke color. It is possible to extract the part of a character where it crosses the reference lines and even goes outside of the rectangle.

One example is illustrated in Fig. 11. The extracted rectangle containing a focus item is shown in (a) while the corresponding wavelet sub-images at levels 1, 2 and 3 are listed in (b1), (b2) and (b3), respectively. To make them visually comparable, the images are shown after adjusting them so that they are the same size. The extracted item strokes are shown in (c).

Most current methods of extracting strokes suppose that the strokes have a fixed color. In our method, strong responses to the strokes are obtained in the extracted rectangle, no matter what the colors of the background and the strokes are, the only requirement being that their colors are different and that the background in the small rectangular region is not too complicated. The example of strokes extraction shown in Fig. 11 has two different colors and it is easy to extract both items successfully. In practice, the first and second steps can be merged by using only one level sub-images, especially when the rectangle is small. However, using the multi level sub-images for a large rectangular region may save computing time.

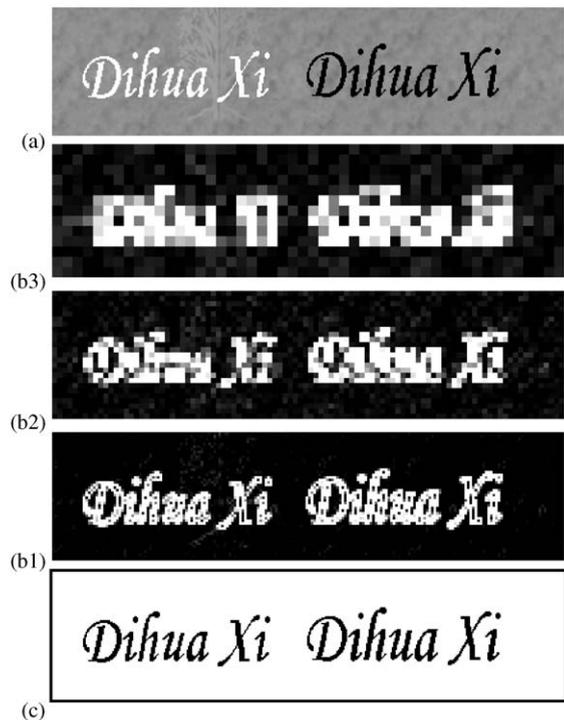


Fig. 11. Example of the item strokes extraction from an extracted rectangle. (a) Extracted rectangle according to the form structure. (b1–b3) Related responses of  $f_{LH}^2 + f_{HL}^2$  in sub-images at levels 1–3. (c) Extracted fill-in characters.

#### 4. Experimental results and analysis

The performance of the proposed algorithm was tested using a form image database containing 368 document images. We also demonstrated the superiority of our algorithm by comparing its performance with that of the well-known

method using the Hough transform and with that of another novel method based on the orthogonal Daubechies wavelet [1]. Most of the other methods of form processing are designed for simple binary images, and thus are difficult to apply to complicated forms. Therefore, it is difficult to compare the proposed method with other methods.

The experiments were conducted using Visual C++ 6.0 on Microsoft Windows 2000. The form database includes several categories of forms, such as sales slips, airplane tickets, bank checks, telephone bills, and forms used in post offices, schools, supermarkets, etc. The original form documents were imported using an optical scanner and converted to monochrome gray scale images. The scanning resolution range was from 200 to 600 dpi (dots per inch). The width or height of the images in the database varies from 300 to 7000 pixels. Most of the form images in the database are not skewed, but there are some images with a slight skew (less than  $2^\circ$ ).

To illustrate the results of our system for those forms which are not skewed, three examples are shown. To save space, the original images are not shown in this paper, except for the airplane ticket, but they are very similar to the LL sub-images. The parameters concerning the technical details are illustrated in Table 1 and a comparison of the processing time with that of the other two methods is given in Table 2.

The first example is used to explain our algorithm which is described in Section 3. The original image is a form used in Korean banks and the image has been contaminated. The form structure extracted from Fig. 8 is shown in Fig. 10(b) and one example item (customer name) is proposed in Fig. 11.

The second example, an airplane ticket, is shown in Fig. 12. The original document (Fig. 12a) is very complicated because it contains not only printed texts in different

colors, fill-in texts and a complicated colorful background, but also has an intricate form structure with reference lines of different lengths. Moreover, there is some overlapping here and there. However, strong responses were obtained to the lines, and clutter was kept to a minimum in the sub-images (Fig. 12b). The extracted forms and some items are shown in Fig. 12c and d, respectively.

Another example is a KS card sales slip shown in Fig. 13. The original image of the sales slips does not have much color information, but contains some dashed lines and the edges of the rectangle in the bottom left are not normal lines.

We also prepared an example (Fig. 14) to show the application of our method to forms with a slight skew angle. In this example, the skew angle in the image is  $1^\circ$ . When applying our method to the skewed form, the shortest horizontal  $L_{min}^H$  and vertical  $L_{min}^V$ , in the step involving line extraction by strip growth, should not be too large, because the length of the strong response for one single line should not be too long. Therefore, the reference lines are easily confused by the presence of large characters. It would be possible to develop an algorithm to overcome this problem using the properties of the responses to the skewed lines. However, this topic is outside the scope of this paper and will not be discussed here. We will be preparing another paper to discuss the application of the wavelet to skewed form documents in the near future. However, the method proposed in this paper works well in the case of form images with a slight skew (less than  $2^\circ$ ).

According to our experimental results, the reference lines can be extracted even when there is some overlap. We also discussed the issue of item extraction from an extracted rectangular field in Section 3.3. The items can be extracted even when overlapping occurs. It is also possible to extract the part of an item that goes outside of the rectangle.

Table 1  
Detailed parameters used in the examples

	$L_{min}^H$ (pixels)	$L_{min}^V$ (pixels)	Max decomposition level	Extracted lines
Fig. 8, 400 dpi	20	15	2	23 horizontal, 26 vertical
Fig. 12, 600 dpi	20	30	3	33 horizontal, 34 vertical
Fig. 13, 200 dpi	70	30	3	16 horizontal, 17 vertical
Fig. 14, 300 dpi	20	15	3	29 near horizontal 19 near vertical

Table 2  
Comparison of the extraction times for the three methods

	Image size (pixels)	Our method (s)	Method in Ref. [1] (s)	Hough transform (s)
Fig. 8, 400 dpi	1048 × 743	0.86	3.10	34.74
Fig. 12, 600 dpi	2356 × 969	2.12	6.23	74.90
Fig. 13, 200 dpi	714 × 1098	0.78	2.94	30.81
Fig. 14, 300 dpi	1725 × 1160	2.08	4.47	60.53

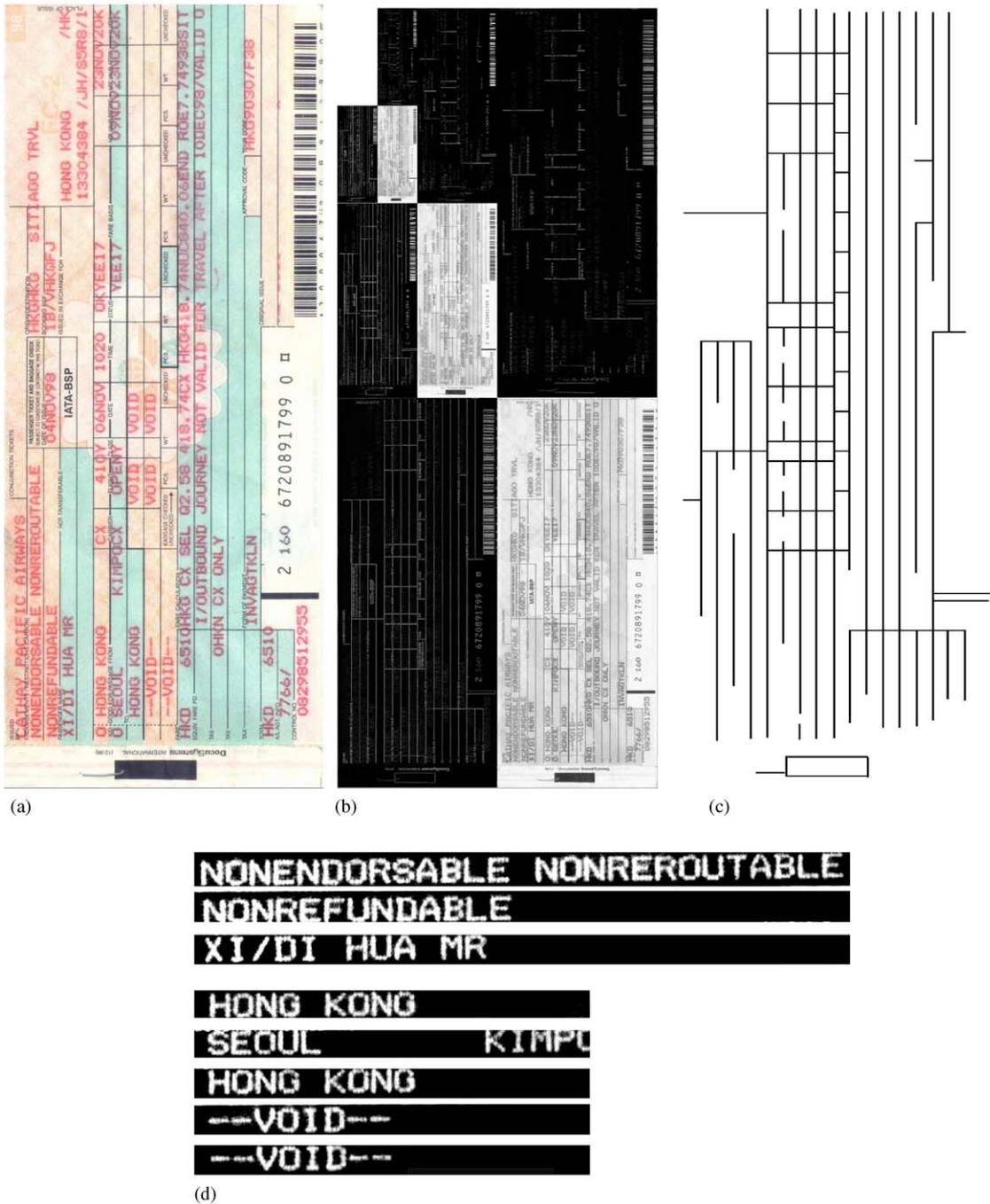


Fig. 12. An example of an airline ticket. (a) The original image; (b) multiresolution G-C wavelet sub-images; (c) extracted form; (d) some extracted items.

This can be done based on the variation in the responses at the crossing point and then tracing the extracted part outside of the rectangle based on the color or intensity information of the extracted strokes. However, this is not the main purpose

of this paper; therefore, please refer to other papers for more information on this topic [7,8].

Please note that all of the examples shown in the paper are displayed in reduced size, the original sizes of the test

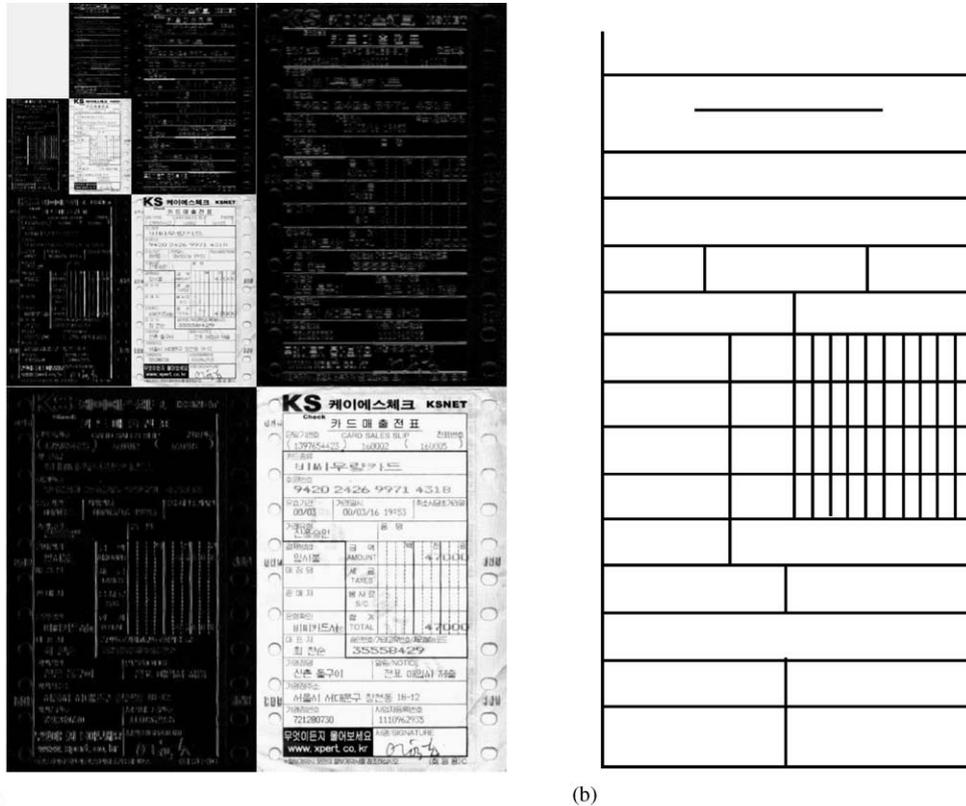


Fig. 13. An example of a KS card sales slip. (a) Multiresolution G–C wavelet sub-images; (b) extracted form; (c) some extracted items.

images being listed in Table 2. The processing times are compared with those of the other methods and the results for the examples are also shown in Table 2. The extraction time depends on the complexity of the input image. In general, the processing time required by our system is less than half that of the method described in Ref. [1] and only about 5% of that of the method based on the classical Hough transform.

5. Conclusions

In this paper, we proposed a pseudo wavelet based system for the extraction of reference lines, as well as the items, from complicated gray-level form document images. Our methodology can overcome the problems of slow computing and large memory requirements associated with the widely used HT based line extraction methods. Compared to the novel method based on the 2D Daubechies orthog-

onal wavelet described in Ref. [1], the Gaussian–Coiflet pseudo wavelet described in this paper is better adapted to the problem of line extraction, mainly because of the use of an adjustable rectangular support in place of the traditional square support. The use of the multiresolution wavelet decomposition method, and its implementation using 2 times 1D wavelets, makes our system much faster than the method described in Ref. [1]. In this paper, we not only presented the algorithm used for line extraction, but we also proposed a complete system of form document processing, including both the extraction of reference lines, using a strip growth method, and the extraction of the items of interest, based on the LH and HL wavelet sub-images.

The experimental results show that the occasional cases of false extraction are mainly due to the following situations.

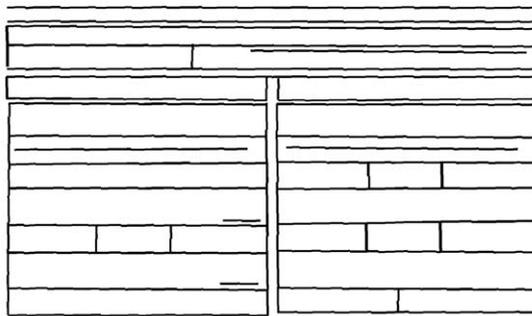
- The length of the reference lines is too short for them to be distinguished from the text or from filled in strokes



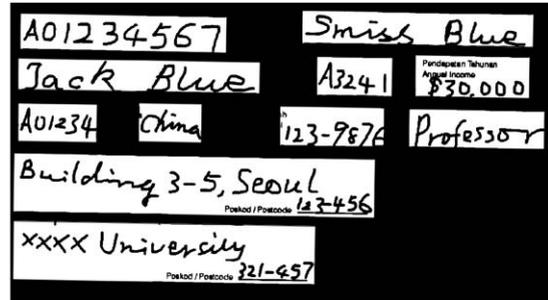
(a)



(b)



(c)



(d)

Fig. 14. An example of skew form image processing. (a) The original image; (b) multiresolution G-C wavelet sub-images; (c) extracted form; (d) some extracted items.

whose length is bigger than or similar to that of the shortest reference lines.

- The scanning rate is too low, which leads to the reference lines being too thin and the responses for the area around them not being strong enough.
- When the original image is skewed at an angle which is not negligible, preprocessing should be performed to provide skew correction.

Furthermore, multiresolution analysis may only be applicable to cases in which the source images are large. In the case of small form images, wavelet decomposition only needs to be performed at level 1. In addition, future works will be concentrated on the estimation of skew angles based on the wavelet sub-images.

## 6. Summary

Form document analysis is one of the most essential tasks in document analysis and recognition. One fundamental and crucial task is the extraction of the reference lines which are contained in almost all form documents. Several approaches

to this problem have been developed in recent years. Most of the studies performed so far, however, were done in connection with the binary images of the documents and, therefore, cannot be readily applied to more complicated image formats. More recently, some researchers have made effort to progress in the processing of gray form images. The HT is a popular method for generic line extraction, but the computing time and storage space associated with this technique are too onerous for it to be applied to the processing of real complicated forms. A novel algorithm based on orthogonal wavelet decomposition has also been developed by Tang et al. [1], but it has some innate defects mainly because of the square support which is used.

In this paper, we propose a pseudo wavelet based system for the extraction of reference lines from complicated gray-level form document images. Our methodology can overcome the problems of slow computing and large memory requirements, which are associated with the widely used HT based line extraction methods. Compared to the novel method based on the 2D Daubechies orthogonal wavelet, the Gaussian-Coiflet pseudo wavelet described in this paper is better adapted to the problem of line extraction, mainly because it employs an adjustable rectangular support in place

of the traditional square support. The use of the multiresolution wavelet decomposition method, and its implementation using 2 times 1D wavelets, makes our system faster than Tang's 2D Daubechies wavelet based method.

Another important problem which is commonly encountered in automatic form document processing is the extraction of item strokes. Variations in the intensity (color) of these fill-in strokes and their complicated background makes this problem all the more difficult. Most current methods suppose that the strokes have a fixed color. In this paper, we also present a method of extracting these items, which is independent of the intensity or color of the strokes, using the previously extracted reference lines and the multiresolution wavelet sub-images. In our method, strong responses to the strokes are obtained in the extracted rectangle, which can be estimated based on the form structure and the extracted reference lines, irrespective of the color of either the background or the strokes. The only requirement is that the colors corresponding to the strokes and background are different and that the background in the small rectangular region is not too complicated. As shown in the experiments, the proposed algorithms demonstrate high performance and fast speed for complicated form images. This system is also effective for form images with a slight skew (less than  $2^\circ$ ).

### Acknowledgements

This work was supported by the Creative Research Initiatives of the Korean Ministry of Science and Technology. The authors would like to thank Prof. Yuan Y. Tang for helping us with various problems and for his valuable comments. They would also like to thank the anonymous reviewers for their careful reading of the manuscripts and for their valuable comments.

### References

- [1] Y.Y. Tang, H. Ma, J. Liu, B. Li, D. Xi, Multiresolution analysis in extraction of reference lines from documents with gray level background, *IEEE Trans. Pattern Anal. Mach. Intel.* 19 (8) (1997) 921–926.
- [2] R.G. Casey, D.R. Ferguson, K.M. Mohiuddin, E. Walach, Intelligent forms processing system, *Mach. Vision Appl.* 5 (3) (1992) 143–155.
- [3] S.L. Taylor, R. Fritzson, J.A. Pastor, Extraction of data from preprinted forms, *Mach. Vision Appl.* 5 (5) (1992) 211–222.
- [4] K.-C. Fan, J.-M. Lu, L.-S. Wang, H.-Y. Liao, Extraction of characters from form documents by feature point clustering, *Pattern Recognition Lett.* 16 (1995) 963–970.
- [5] Y. Ishitani, Flexible and robust model matching based on association graph for form image understanding, *Pattern Anal. Appl.* 3 (2) (2000) 104–119.
- [6] J.-L. Chen, H. Lee, Field data extraction for form document processing using a gravitation-based algorithm, *Pattern Recognition* 34 (2001) 1741–1750.
- [7] B. Yu, A.K. Jain, A generic system for form dropout, *IEEE Trans. Pattern Anal. Mach. Intel.* 18 (1996) 1127–1134.
- [8] X. Ye, M. Cheriet, C.Y. Suen, A generic method of cleaning and enhancing handwritten data from business forms, *Int. J. Doc. Anal. Recogn.* 4 (2001) 84–96.
- [9] L.Y. Tseng, R.C. Chen, Recognition and data extraction of form documents based on three types of line segments, *Pattern Recognition* 31 (1998) 1525–1540.
- [10] J. Liu, A. Jain, Image-based form document retrieval, *Pattern Recognition* 33 (3) (2000) 503–513.
- [11] H. Zhao, B. Liu, Z. Jiang, T. Ostgathe, Global-local-global method for logical structure extraction of form document image, *J. Electron. Imaging* 9 (2000) 296–304.
- [12] C.F. Lin, C.Y. Hsiao, Structural recognition for table-form documents using relaxation techniques, *Int. J. Pattern Recogn. Artif. Intel.* 12 (7) (1998) 985–1005.
- [13] C. Cracknell, A.C. Downton, A colour classification approach to form dropout, *Proceedings of the International Workshop on Frontiers of Handwriting Recognition*, Vol. 6, Taejon, Korea, 1998, pp. 485–495.
- [14] M. Okada, M. Shridhar, Extraction of user entered components from a personal bank check using morphological subtraction, *Int. J. Pattern Anal. Artif. Intel.* 11 (5) (1997) 699–715.
- [15] J.-L. Chen, H.-J. Lee, An efficient algorithm for form structure extraction using strip projection, *Pattern Recognition* 31 (1998) 1353–1368.
- [16] X. Ye, M. Cheriet, C.Y. Suen, K. Liu, Extraction of bank check items by mathematical morphology, *Int. J. Doc. Anal. Recogn.* 2 (1998) 53–66.
- [17] S. Mallat, Multiresolution approximations and wavelet orthonormal bases of  $L^2(\mathbb{R})$ , *Trans. Am. Math. Soc.* 315 (9) (1989) 69–87.
- [18] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, 1998.
- [19] I. Daubechies, Orthonormal bases of compactly supported wavelets, *Commun. Pur. Appl. Math.* 41 (11) (1988) 909–996.

**About the Author**—DIHUA XI received his BS degree in Mathematics from Sichuan Normal University and his MS degree in applied mathematics from Sichuan University of China in 1986 and 1995, respectively. Currently, he is a Ph.D. candidate at the Department of Visual Information Processing, Korea University, Korea. From 1995 to 1998, he served as an assistant professor and lecturer at Sichuan University, China. He served as a visiting scholar at Hong Kong Baptist University in 1998. His present research interests include wavelet analysis, fractal geometry, support vector machines and their applications in document image analysis, computer vision and the pattern recognition related fields.

**About the Author**—SEONG-WHAN LEE received his BS degree in Computer Science and Statistics from Seoul National University, Seoul, Korea, in 1984, and his MS and Ph.D. degrees in computer science from KAIST in 1986 and 1989, respectively. From February 1989 to February 1995, he was an assistant professor in the Department of Computer Science at Chungbuk National University, Cheongju,

Korea. In March 1995, he joined the faculty of the Department of Computer Science and Engineering at Korea University, Seoul, Korea, as an associate professor, and he is now a full professor. Currently, Dr. Lee is the director of the National Creative Research Initiative Center for Artificial Vision Research (CAVR) which is supported by the Korean Ministry of Science and Technology. He was the winner of the Annual Best Paper Award of the Korea Information Science Society in 1986. He obtained the First Outstanding Young Researcher Award at the Second International Conference on Document Analysis and Recognition in 1993, and the First Distinguished Research Professor Award from Chungbuk National University in 1994. He also obtained the Outstanding Research Award from the Korea Information Science Society in 1996. He has been the co-Editor-in-chief of the International Journal on Document Analysis and Recognition since 1998 and the associate editor of the Pattern Recognition Journal, the International Journal of Pattern Recognition and Artificial Intelligence, and the International Journal of Computer Processing of Oriental Languages since 1997. He was the Program co-chair of the Sixth International Workshop on Frontiers in Handwriting Recognition, the Second International Conference on Multimodal Interface, the 17th International Conference on the Computer Processing of Oriental Languages, the Fifth International Conference on Document Analysis and Recognition, and the Seventh International Conference on Neural Information Processing. He was the workshop co-chair of the Third International Workshop on Document Analysis Systems and the First IEEE International Workshop on Biologically Motivated Computer Vision. He served on the program committees of several well-known international conferences. He is a fellow of IAPR, a senior member of the IEEE Computer Society and a life member of the Korea Information Science Society and the Oriental Languages Computer Society. His research interests include pattern recognition, computer vision and neural networks. He has published more than 150 publications in these areas in international journals and conference proceedings, and has authored five books.