IJDAR (1999) 2: 53–66

International Journal on IJDAR
Document Analysis and Recognition


# Extraction of bankcheck items by mathematical morphology

**Xiangyun Ye[1,2], Mohamed Cheriet[1,2], Ching Y. Suen[1], Ke Liu[1]**

[1] Centre for Pattern Recognition and Machine Intelligence, Concordia University, Suite GM 606, 1455 de Maisonneuve Blvd. West, Montreal, Québec H3G 1M8, Canada
[2] Laboratorie d'imagerie, de vision et d'intelligence artificielle, École de Technologie Supérieure de l'Université du Québec, 1100, rue Notre-Dame Ouest, Montréal, Québec H3C 1K3, Canada

Received June 16, 1998 / Revised June 18, 1999

**Abstract.** This paper presents a technique for extracting the user-entered information from bankcheck images based on a layout-driven item extraction method. The baselines of checks are detected and eliminated by using gray-level mathematical morphology. A priori information about the positions of data is integrated into a combination of top-down and bottom-up analyses of check images. The handwritten information is extracted by a local thresholding technique and the information lost during baseline elimination is restored by mathematical morphology with dynamic kernels. A goal-directed evaluation of the extraction approaches is proposed, and both qualitative and quantitative analyses show noticeable advantages of the proposed approach over the existing approaches.

**Key words:** Check image processing – Baseline elimination – Local thresholding binarization – Gray level mathematical morphology – Goal-directed evaluation

## 1 Introduction

During the past five years, automatic bankcheck processing has become a popular topic in document analysis, and is becoming one of the most promising commercial applications of handwriting recognition [1]. Benefitting from the availability of ever-growing CPU power, the development of multiple CPU systems makes it possible to process huge amount of checks at high speed, and thus relieve people from intensive and tedious jobs. However, in spite of great progress in the recognition of unconstrained handwriting made in recent years, bankcheck processing still presents an important challenge to the scientific communities working in the field. To recognize the items on a bankcheck, it is necessary not only to locate the items, to remove the boxes and lines, to separate the user-entered items from the background and noise, but also to split touching characters, to merge broken strokes, and to tackle poor-quality handwritings. Each step of the check processing involves fragile operations by which a small mistake can result in a wrong final answer.

The lack of standards for checks is one of the main problems that causes great difficulties for the automation of the process. Bankchecks differ not only in background, but also in type and position of the preprinted and user-entered information. The variability of the size and structure of bankchecks, together with the intrinsic complexity of the character recognition problem, makes the development of general algorithms and strategies for bankcheck processing extremely difficult. Up to now, several solutions have been proposed by research groups in different countries including Italy [2], Canada [1], France [3–5], Brazil [6], P. R. China [7], U.S.A [8–10], Japan [11], etc., and great efforts have been contributed to the development of highly reliable industrial applications with high performance.

The information fields to be filled in by the customers generally consist of the legal amount (LA in brief), courtesy amount (CA in brief), date and signature, which are mostly guided by preprinted long lines called baselines. The checks are often deliberately designed with specific styles, colorful and complicated backgrounds, causing difficulties in both the extraction and recognition of the items. Clearly, the first and basic step of automatic reading of checks is item extraction. Only after the items have been accurately extracted, is the system able to recognize the amounts and other information on the checks. Nevertheless, these issues have rarely been thoroughly studied and most published systems generally rely on ad hoc strategies or assume that the items are positioned at known locations over neat backgrounds. Table 1 lists the item extraction methods used by some recent systems and their recognition results. However, since in some cases the quality of the data and the experimental conditions were not described, and there is no standard database to test the check processing systems, it is not possible to compare the results. The performance of the systems based on different databases is listed to illustrate

*Correspondence to*: X. Ye

**Table 1.** Some recently published systems

| Reference | Raw image | item extraction | Binarization | Baseline removal | Test set | Rec.(%) | Rej.(%) | Err.(%) |
|---|---|---|---|---|---|---|---|---|
| G. Dimauro et al [2] | $G^*$ | Bank database | Hill-clustering | Controlled thinning & Math. Morph. | 1500 | 96.3 | 0 | 3.7 |
| S. Knerr et al [3] | $B^*$ | Bottom-up | - | Local features analysis | 32000 | 70.6 | 24.3 | 5.1 |
| L. L. Lee et al [6] | G | Standard layout | Fixed background subtraction | - | 121 | 72.7 | 0 | 27.3 |
| L. Heutte et al [4] | B | Manual or automatic location | - | HT† | 3374 | 52.94 | 26.01 | 21.05 |
| M. Leroux et al [5] | B | Guideline directed | - | Local vicinity analysis | 10000 | 60 | 28 | 12 |
| G. Dzuba et al [8] | B | Rule based | - | HT | 5000 | 75 | 0 | 25 |
| G. Kim et al [9] | B | Guideline directed | - | Run-length CCP‡ analysis | 235 | 43.8 | 56.2 | 0 |
| K. Liu et al [12] | G | Layout directed | Edge detection | HT | 499 | 67.35 | 0 | 32.65 |
| M. Okada et al [11] | G | Reference database | - | Math. Morph. subtraction | - | - | - | - |
| A. Agarwal et al [10] | G | Bottom-up | Global thresholding | Width-height ratio analysis | - | - | - | - |

* — G means Gray-level image and B means Binary image
† — HT refers to Hough Transform
‡ — CCP refers to connected components

the state-of-art instead of making quantitative comparisons among different methods.

Some of the systems [3–5]were based on binary images obtained from checks with simple backgrounds, and some other systems were based on gray images and bank reference databases [6,11]. The baselines, acting as guides for users to fill in the checks, are extracted and eliminated by line detection methods including least square fitting [12], Hough Transform [4,8], and wavelet [13], which have high computational complexity and require large memory storages. Although some of the above systems acquired relatively high performance on checks with simple, fixed style backgrounds, problems exist when real-life checks with varying styles and complex backgrounds are encountered. The major problems are caused by poor quality of binarization in complex backgrounds and the noise remaining after the baselines are eliminated. The reliance upon bank reference databases makes some systems inapplicable to the processing of multi-bank checks, and the input of binary images makes certain other systems unable to discriminate handwritings from the textured and colorful backgrounds, which are the usual cases in real-life applications. The problems are partly solved by Cheriet et al. [14], but still exist when the histogram of handwritings overlaps with that of the background.

In this paper, an item extraction method based on gray-level mathematical morphology and a local thresholding technique is proposed. The method is applied to gray-level images instead of binary images because of the following reasons:

- A real bankcheck may contain a complex background, which causes difficulty in finding an effective global thresholding method to produce a satisfactory binary image.
- The gray-level mathematical morphology preserves the difference between handwritings and the baselines, while they are treated as same during binary scanning.
- Baselines extracted and eliminated in gray-level images result in a smoother boundary than that in binary images.
- The baselines extracted by gray-level mathematical morphology result in a more accurate description than that extracted by sub-optimal binarization along with binary morphological operations.

The proposed method begins with the extraction and elimination of the baselines by gray-level mathematical morphological operations, then applies a local thresholding technique to separate the handwritten strokes from the background. With the layout description obtained by analyzing the extracted baselines, the candidate connected components are detected and grouped together to extract the items. The extracted items can be used as the input of subsequent recognition modules. An overview of the system is given in Sect. 2; details of the proposed approach is discussed in Sect. 3; experimental results as well as qualitative and quantitative analyses are given in Sect. 4.

## 2 System overview

Bankcheck image processing systems aim at recognizing the amount (legal amount and courtesy amount), and validating the date and signature. The item extraction

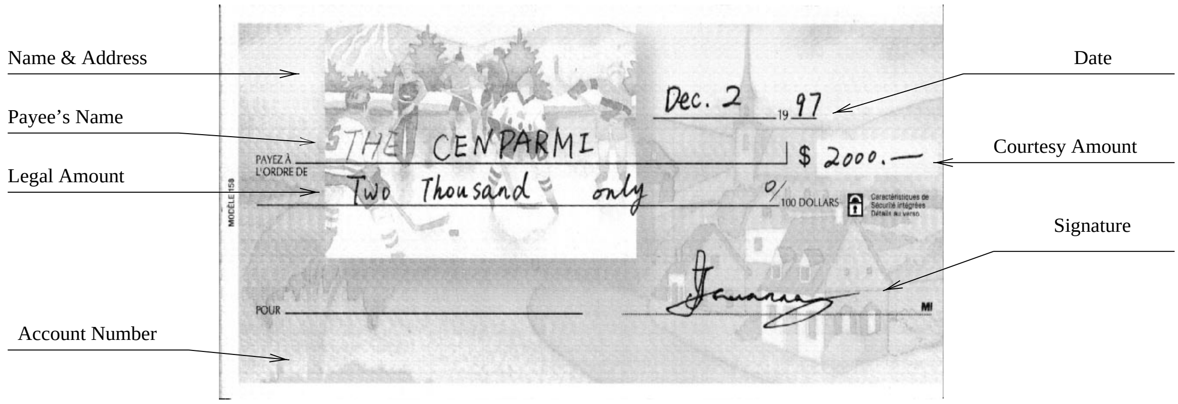
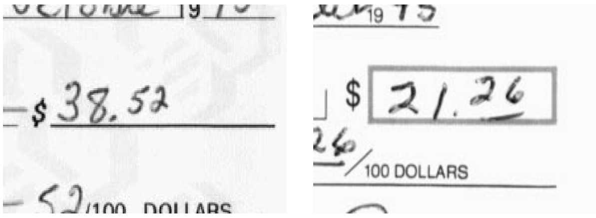



**Fig. 1.** Example of a typical Canadian bankcheck



a b

**Fig. 2a,b.** Different styles of courtesy amount. Courtesy amount guided by **a** baseline; courtesy amount guided by **b** rectangular box

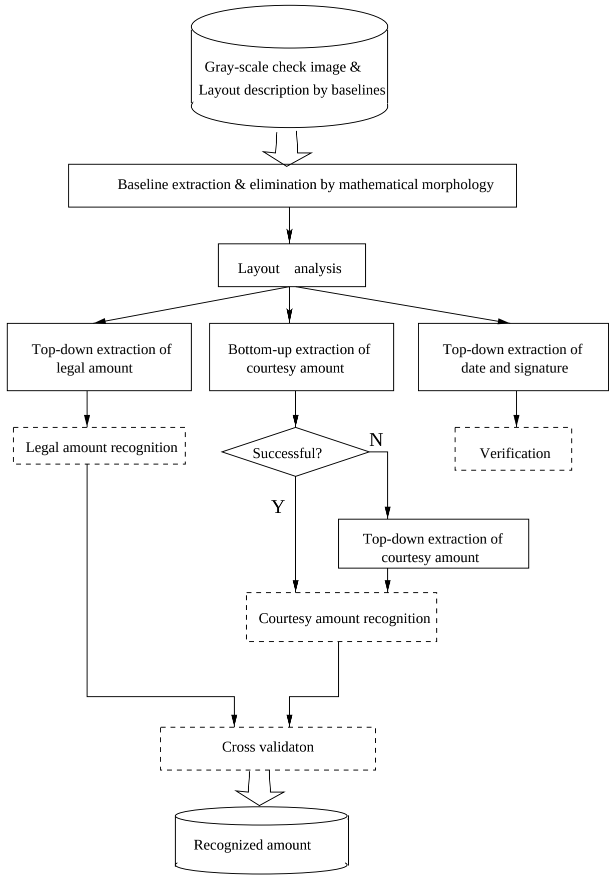



**Fig. 3.** System overview

procedure analyzes the layout or logic structure of the input check images, and thus provides meaningful blocks of amounts or other items to the recognition modules. The input of the system is the scanned check images in gray levels, and the outputs of the bankcheck item extraction procedure are the binarized courtesy amount, legal amount, date, signature, etc. According to the known constraints and a-priori information about check formats, the analysis of check images can be carried out via two approaches, i.e., bottom-up and top-down methods.

For a standard Canadian check as shown in Fig. 1, the legal amount is written above a long line called the legal amount baseline. Above this, is a parallel long baseline called payee baseline. The date is written above two short lines at the same height called date baselines. The signature is written on a lower line called signature baseline. For the courtesy amount, there might be a baseline (Fig. 2a) or a surrounding rectangular box in low contrast (Fig. 2b).

The items guided by baselines can be extracted by detecting each baseline and grouping connected components within the surrounding regions determined by the check layout information. However, there is no guarantee for the existence of the courtesy amount baseline (Fig. 2b), which makes the extraction of the courtesy amount a more complicated problem than that of other items. Figure 3 shows the basic functional modules for the automatic item extraction from bankcheck images. The input of the system is a gray-scale check image and the outputs are binarized items, used as inputs to the subsequent recognition modules. The modules in solid boxes belong to the item extraction module and will be described in the following section.

## 3 Extraction of bankcheck items

### *3.1 Top-down extraction of items*

In a previous system proposed by Liu et al. [12], a novel and generic approach was presented to extract legal and courtesy amounts and dates from Canadian checks by analyzing gray-scale check images directly. Once a check

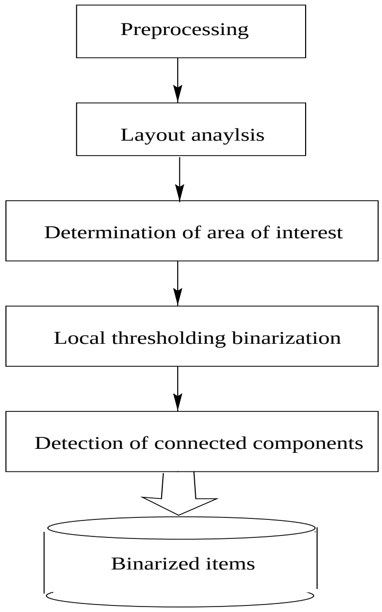



**Fig. 4.** Top-down extraction of items

has been scanned and converted into a gray image, the preprocessing module is applied to produce two more images, i.e., the image of candidate edge points and the image of edge directions, respectively. Based on these two images and the original one, the baselines are extracted by Hough Transform and Least Square Fitting. The system then uses a priori knowledge about Canadian checks to determine the layout description of the check by analyzing these baselines. Once the layout analysis is finished, the system proceeds to extract the legal amount, courtesy amount and date, respectively. In order to extract the items, the system determines the searching and bounding regions which contain them, estimates the gray-level distributions of the related handwritten strokes, and traces out the connected components. After the items are extracted, a final module is applied to remove the intersecting baselines or the bounding rectangular boxes of the courtesy amount.

In this paper, we still use the layout analysis module of [12], except that the extraction and elimination of the baselines are carried out by gray-level mathematical morphology, and global thresholding is replaced by a local thresholding technique described in Sect. 3.4. The modules corresponding to the top-down analysis are shown in Fig. 4, in which "preprocessing" refers to the extraction and elimination of baselines.

In real applications, there exist problems in identifying the courtesy amount baseline when there is only a rectangular box in low contrast, and the currency symbol is damaged by noise. In these cases, the following a priori knowledge about the CA field can be used to extract the courtesy amount:

– If there is a short baseline on the right-hand side of the payee baseline and it has the same height as that of the payee baseline, the CA lies above this short line.

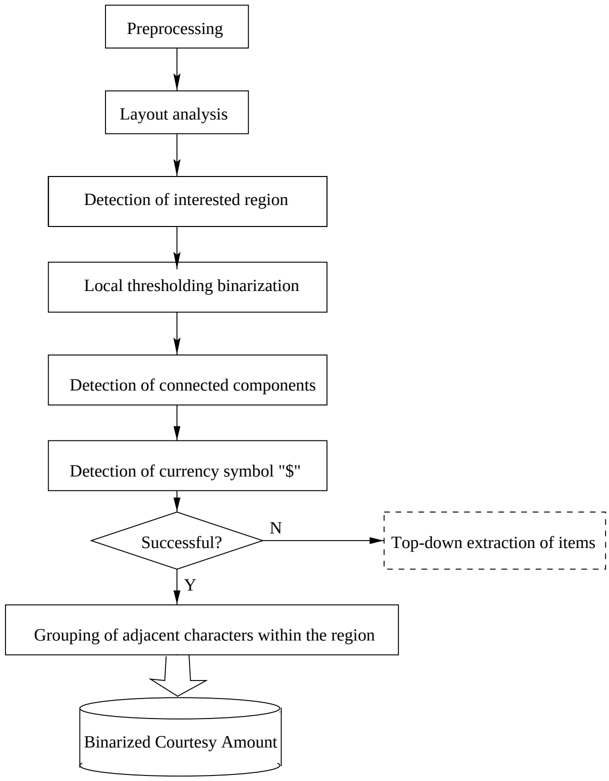



**Fig. 5.** Bottom-up extraction of courtesy amounts

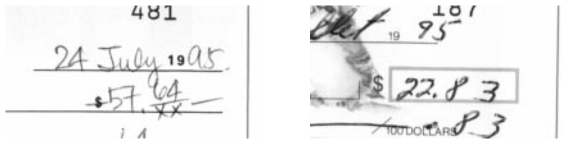
– If the right end points of both legal amount and payee baselines almost reach the right edge of the check, the CA lies above the right part of the payee baseline.
– If the conditions corresponding to the above two rules are not satisfied, the CA may lie on the right-hand side of the legal amount or payee baseline.

### *3.2 Bottom-up extraction of courtesy amounts*

The detection and identification of baselines provide basic layout information of the check images for item extraction. However, different banks print checks in their own document layout designs, which may include elaborate backgrounds and varying styles of courtesy amount guidelines. Meanwhile, there exist some cases when the system fails to detect the broken or absent baselines. Concerning the importance and special difficulty of the courtesy amount over other items, we propose to combine both a top-down method based on baseline extraction and a bottom-up method based on character grouping. Since the bottom-up method provides more precise information about the location of CA and is more robust against noise, the courtesy amounts are first extracted by the bottom-up method as illustrated in Fig. 5, and then extracted by the top-down analysis described in the previous section if the system fails to detect the presence of currency symbol.


a b

**Fig. 6a,b.** Detection of currency symbol and locating of the courtesy amount(bottom-up) **a** Currency symbol detection **b** Connected components grouping




a b

**Fig. 7a,b.** Failure of detecting currency symbol **a** "$" interfered by strokes **b** "$" connected with background

3.2.1 Determination of area of interest

In the bottom-up CA extraction methods, all connected components within a given check image are extracted. Starting from the pixel level, the method proceeds via character parts, characters, words, finally to the amount level. At each hierarchical level, the candidate components are grouped together according to a set of measures such as horizontal collinearity. Since the grouping of all connected components in the whole image is time consuming and inefficient, and all CAs are always located on the right side of either the legal amount baseline or the payee baseline obtained in Sect. 3.3, we limit the location of CA to a certain region decided by the legal amount baseline and payee baseline. The rectangular boxes surrounding CAs are removed by the baseline removal approach described in Sects. 3.3.2–3.3.4, except that the order of horizontal and vertical closings are interchanged in removing vertical lines.

3.2.2 Extraction and grouping of components

For a standard Canadian check, in addition to the fact that the CA field is always located in the upper-right corner of a check, there is always a machine printed currency symbol "$" in front of the CA. After a *Depth First Search* for connected components, the position of "$" can be located by a full-fledged character recognizer. The right side of "$" decides the precise beginning of user-entered CA, and the connected components whose minimal bounding box overlaps with the region decided by "$" can be grouped together to detect the whole string of CA (Fig. 6). Once the grouping process is completed, the CA is represented by a string of isolated or connected characters.

Over 97% CA strings can be extracted by the bottom-up analysis introduced above. When there exist problems with unrecognizable "$" symbols due to complex backgrounds or intersecting handwritten strokes (Fig. 7), the top-down extractor based on baselines and layout analysis is used.

*3.3 Baseline extraction and elimination*

In our approach, the preprinted baselines on checks are used to describe the structure of checks for locating the zones which contain the interesting items. Therefore, the first step of item extraction is to extract all baselines from the check images. In real-life applications, the checks are usually scanned in an upright direction, which simplifies the extraction of baselines. The proposed approach of extracting and eliminating the baselines in bankcheck images can be easily extended to other kinds of forms and documents involving line detection.

3.3.1 Gray-level mathematical morphology

Ever since Serra [15] formalized the theory of mathematical morphology, it has been widely used in solving image processing problems difficult for linear filters. Based on shape, mathematical morphology provides an efficient approach to process digital images. Appropriately used, it tends to simplify image data by preserving their essential shape characteristics and eliminating irrelevancies. The basic mathematical morphological operations are erosion, dilation, opening, and closing. Erosion ($\ominus$) and dilation ($\oplus$) of a gray-level image or function by a flat structuring element (SE in brief) may be defined as the maximal or minimal value of a local window [16]. Opening ($\circ$) and closing ($\bullet$) operations are based on a cascade of erosion and dilation.

A morphological operation can be seen as a detector of certain image features described by SE. The union or intersection of a series of linear openings or closings can be used to extract long-thin features within an image, independently of their orientations. Consequently, in order to extract baselines in bankcheck images, which are mostly horizontal long lines, the following operations are proposed.

3.3.2 Baseline estimation

Assuming that the check image $F = \{f(x,y)\}$ has a set of pixel values $f(x,y) = f_{xy}$ at each pair of discrete coordinates $(x,y)$ (Fig. 8a). We define a horizontal linear SE $H$ with a given length $l$, which corresponds to the length of the shortest baseline feature to be eliminated. Meanwhile, the SE should be large enough so that long handwritten strokes will be preserved.

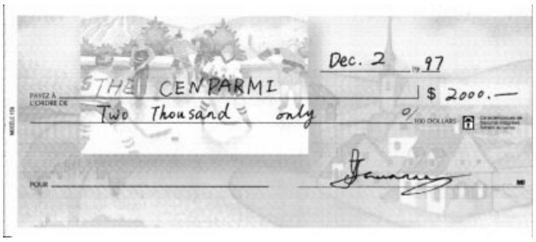
Details other than the long baselines in a bankcheck image $f$ are first smeared out by horizontal closing:

$$F_1 = \{f_1(x,y)\} = \{(F \bullet H)(x,y)\} \quad \forall (x,y) \in F \qquad (1)$$

After the horizontal closing, only long lines and slowly varying backgrounds are preserved in the image $F_1 = \{f_1(x,y)\}$ (Fig. 8b).

Now, we can define a vertical linear SE $V$ with a given height $h$, which corresponds to that of the thickest part of the baselines to be extracted. By applying closing with vertical SE, high frequency variations of gray-level in a vertical direction are removed:

$$F_2 = \{f_2(x,y)\} = \{(F_1 \bullet V)(x,y)\} \quad \forall (x,y) \in F_1 \qquad (2)$$



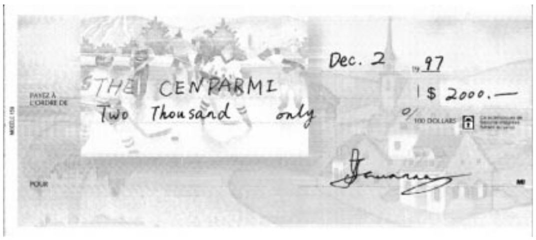

**a** Raw image of bankcheck

**b** Horizontal closing of **a**

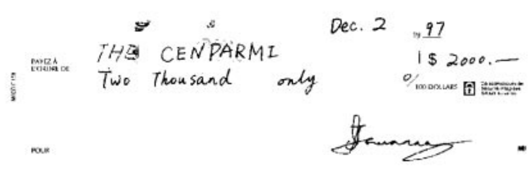


**c** Vertical closing of **b**

**d** Baseline extraction




**e** Baseline elimination




**f** Local thresholding binarization of **e**

**Fig. 8a–f.** Baseline extraction and elimination by gray-level mathematical morphology

The vertical closing operation removes the baselines and results in an approximate description of background (Fig. 8c).

3.3.3 Baseline extraction and elimination

The difference between $F_1$ and $F_2$ is mainly composed of the horizontal long lines which were removed by the vertical closing operation. Therefore, the baselines can be extracted by the following subtraction:

$$F_3 = \{f_3(x,y)\} = \{f_2(x,y) - f_1(x,y)\} \quad \forall (x,y) \in F_2 \quad (3)$$

The precise description of the baselines can be obtained by analyzing $F_3$, in which the baselines exhibit prominence (Fig. 8d). Then the baselines can be eliminated from gray-level bankcheck images by compensating the extracted baselines (Fig. 8e):

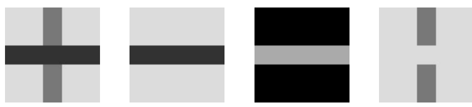
$$F_4 = \{f_4(x,y)\} = \{f(x,y) + f_3(x,y)\} \quad \forall (x,y) \in F_3 \quad (4)$$

3.3.4 Information restoration by dynamic kernels

The proposed method of baseline extraction and elimination may preserve the gray-level difference between the handwritten strokes and the intersected baselines to some extent (Fig. 9). However, for those handwritten strokes having the same gray-level as the intersected baselines, further topological or morphological operation is needed to restore the strokes broken due to the elimination of the baseline (Fig. 10).

a b c d

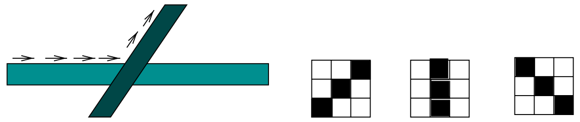
**Fig. 9a–d.** Information preservation handwritten strokes darker than background **a** Raw image **b** Horizontal closing **c** Baseline extraction **d** Baseline elimination



a b c d

**Fig. 10a–d.** Information lost with handwritten strokes lighter than background **a** Raw image **b** Horizontal closing **c** Baseline extraction **d** Baseline elimination

Several approaches have already been reported to restore the broken strokes by binary morphological operations [2,17,18]. Here we propose to restore the broken strokes on gray-scale images instead of binarized images in order to minimize the information loss. As proposed by Cheriet et al. [19], the purpose of the morphological closing operation is to propagate the intensity of the background and the intersecting handwritten strokes over the baselines. With a properly selected structuring element $D$, the strokes that intersected the baselines can be preserved. In fact, in the cases where $D$ has the same orientation as that of the handwriting at the in-



a Stroke tracing b Dynamic kernels

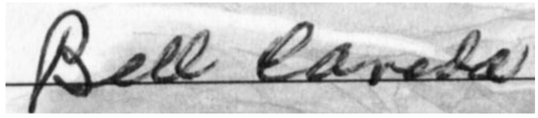
**Fig. 11a,b.** Information restoration by dynamic kernels

tersecting points, the lost information can be completely restored. To avoid the time-consuming topological process, the broken strokes in each possible orientation are restored by applying the mathematical morphology with dynamic kernels[20] within the regions of baselines obtained in Sect. 3.3.3:

$$F_5 = \{f_5(x,y)\} = \begin{cases} f_4(x,y) & \text{if } f_3(x,y) > T \\ (F_4 \bullet D_k)(x,y) & \text{otherwise} \end{cases} \quad (5)$$

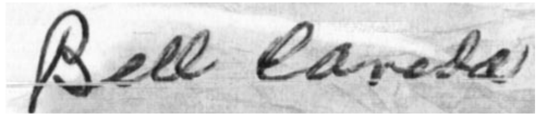
In (5), $T = Ostu(F_3)$ means threshold for baseline image obtained by Ostu's algorithm [21], and $D_k, k = 0..K$ are kernels in different orientations. The dynamic kernel $D_k(k \in [0,..,K])$ is chosen according to the local orientation of the intersecting strokes, which is obtained by tracing the detected baselines. The pixels with maximum gradient value among the neighbourhoods are examined and a stroke is detected when there is a sharp change in the gradient value. The orientation of the intersecting stroke is thus calculated from the gradient direction. Consequently, the broken strokes are restored by closing operation with the respective dynamic kernel. In our experiments, three typical kernels are chosen (Fig. 11). Stroke tracing and restoration are applied to a narrow region decided by the baselines, so that there was little effect on the processing speed of the entire system.

An example of information restoration by dynamic kernels is shown in Fig. 12. Figure 12a is a raw image on which the handwritten characters intersect with the baseline. Estimated by a series of morphological operations, the baseline is extracted and shown in Fig. 12b. The broken strokes due to the elimination of baselines (Fig. 12c) are restored by morphological operations with dynamic kernels, shown in Fig. 12d. In the subsequent section, it will be demonstrated that the information restoration relieves the system from post-processing of repairing the broken strokes.

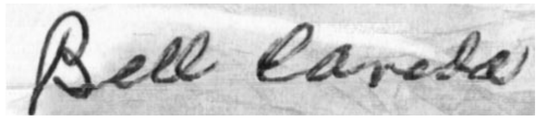


a Raw image

b Baseline extraction



c Baseline elimination



d Morphological restoration

**Fig. 12a–d.** Examples of restoration by dynamic kernels

### *3.4 Local thresholding binarization*

Bankchecks may contain different backgrounds composed of pictures printed in various colors and intensities. To isolate the handwritten information from the complex background, a local thresholding binarization based on Bernsen's method [22] is used. A recent comparative study of binarization methods for document images was given by Trier et al. [23,24]. The eleven most promising locally adaptive algorithms were studied. Eight of the algorithms use explicit thresholds or threshold surfaces, while the other three search for printed pixels after having located the edges. Trier and Taxt concluded that Niblack's [25] and Bernsen's methods along with the post-processing step proposed by Yanowitz and Bruckstein [26] were the fastest and best ones based on a set of subjective and goal-directed criteria. The post-processing step here is used to improve the binary image by removing "ghost" objects, and it requires the average gradient value at the edge of each printed object to be calculated. The objects having an average gradient below a fixed threshold $T_p$ are labeled as misclassified and are therefore removed.

#### 3.4.1 Calculation of local threshold

Although satisfying results can be obtained for a certain class of images with similar gray-level distributions, difficulty arises in finding a good post-processing method for a wide range of images, which is common in bankcheck analysis systems. In the literature, the important parameter $T_p$ was usually selected manually by trial and error. However, it is almost impossible to determine a fixed value for a set of blind test images. As one of the solutions, we propose to facilitate the selection of $T_p$ by estimating the distribution of local contrast, which is an indicator of the property of the input image.

We have the following basic understanding about the characters to be extracted:

- The characters occupy a lower gray-level than local neighborhoods.
- The gray-level variation within characters is limited.
- The characters are thin, connected components whose stroke widths are smaller than a certain limit.
- The boundaries of characters have higher local contrast than that of backgrounds.

The above understandings do not exclude the possibility of overlapping gray-level distribution between characters and backgrounds, which is commonly encountered by a variety of practical systems. According to our observation, the "ghost" phenomena or broken characters are

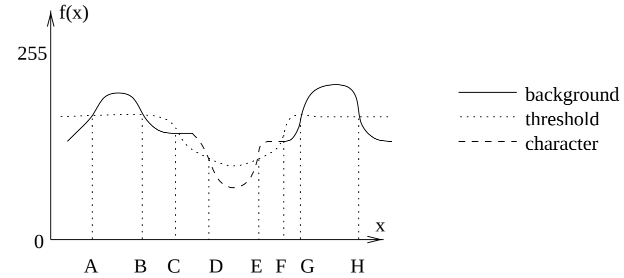



**Fig. 13.** Reasons for "ghost" phenomena

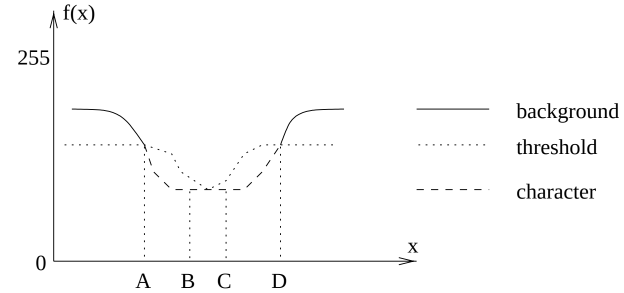



**Fig. 14.** Reasons for broken strokes

usually caused by sudden changes of the foreground or background. As shown in Fig. 13, which is a 1D demonstration of image $f(x)$, the dotted line indicates the threshold for each pixel calculated by Bernsen's method. Pixels between $D$ and $E$ belong to real handwritten strokes, while others belong to the background. However, the background pixels within $BC$ and $FG$ are classified as characters due to the sudden change of background within $AB$ and $GH$. Similarly, the case of broken strokes is shown in Fig. 14, in which pixels between $A$ and $D$ belong to handwritten strokes. The stroke within $BC$ will be binarized into the background due to greater width than the size of the local window and equal gray-levels with the surrounding pixels. To solve the "ghost" problem and broken strokes by Bernsen's method, we propose to eliminate the false detections by thresholding local contrasts.

- For each pixel $(x, y)$ in $F$, a threshold value $T(x, y)$ is chosen:

  $$T(x, y) = (F_{max}(x, y) + F_{min}(x, y))/2 \tag{6}$$

  which is the mean of the highest gray-level value $F_{max}(x, y)$ and the lowest gray-level value $F_{min}(x, y)$ in a square $r*r$ neighborhood centered at $(x, y)$. The window size $r$ is chosen according to an approximation of the stroke width.
- The local contrast image is obtained by calculating the difference between the local minimum and maximum:

  $$C = \{c(x, y)\} = F_{max}(x, y) - F_{min}(x, y) \tag{7}$$

  A local window is considered as consisting of only one class if the contrast is lower than a threshold $t = Ostu(C)$. In the context of document image binarization, where wide printed areas rarely occur, the regions with a contrast lower than the threshold $t$ can be regarded as backgrounds. Actually, the difference between the centre pixel value and the mean of highest and lowest values in the local window is similar to a second-order derivative.
- Binarization is carried out by taking both the threshold of gray-level and the threshold of local contrast into consideration. To obtain a binarized image $B = \{b(x, y)\}$ in which $b(x, y) = 1$ stands for foreground pixels and $b(x, y) = 0$ stands for background pixels, the following algorithm is applied:

  $$B = \{b(x, y)\} = \begin{cases} 1 \text{ if } f(x, y) < T \textit{ and } c(x, y) > t \\ 0 \text{ otherwise} \end{cases} \tag{8}$$

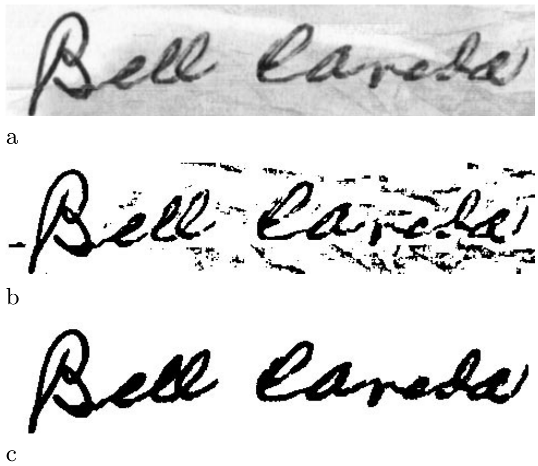



**Fig. 15a–c.** Local thresholding technique **a** Raw image **b** "Ghost" phenomena **c** Proposed method

An example of the local thresholding binarization is shown in Fig. 15, in which Fig. 15a is the same as Fig. 12d. Compared with the binarized image from Bernsen's method (Fig. 15b), the noise is suppressed and the characters are enhanced by our method (Fig. 15c).

### 3.4.2 Fast algorithm of extrema filtering

The two basic operations of gray-level mathematical morphology are dilation and erosion, which can be treated as the maximum and minimum filters in processing with flat structuring elements (SE in brief) [15]. In general, however, the computation time of morphological filters for large kernels becomes an impediment to online applications. Meanwhile, the local thresholding technique described in Sect. 3.4.1 also uses both minimum and maximum filters to calculate the binarization threshold for each pixel. Since the size of a check image is usually very large, and both dynamic thresholding and morphological operations are used in the item extraction module, the computational time of the basic image processing steps becomes an important factor in developing a real-time system. In the classic algorithm of local extrema filtering, the extrema are obtained by comparing each pixel in the local window consecutively, which makes the time cost proportional to the square of the radii of local windows. When we eliminate the baselines from the check images by morphological operations, some long linear SEs are used. Therefore, a fast algorithm is thereby greatly needed to overcome the high consumption time of the classic algorithm.

Since maximum/minimum filters are the special cases of rank-order filters, we begin our algorithm with rank-order filters which sort every pixel in a local window by a complete non-increasing or non-decreasing order. Without losing generality, we consider the realization of maximum filtering. Let the candidate pixels be accessed in a horizontal scanning fashion. When moving from one candidate to the next, the window for the current candidate can be obtained by deleting the leftmost pixel of the previous window and appending a new pixel to the right. While finding the maximum in the current window, the rank of the leftmost pixel is dropped from the ranked gray-levels of the previous window, and the new pixel is inserted using the principle of merge sort. However, notice that when a window slides to find the local maximum, the larger element of $\{f(i)\}$ on the right will screen out the smaller element on the left. Therefore, we do not need to keep the complete order of the window. Since only the maximum is needed by sliding the window, the following fast algorithm can be derived:

Set up an array $T$ with a variable length. The largest possible length is $2r+1$ where $r$ is the radius of a symmetric structuring element. The elements of array $T$ are the indices of pixels in the current observed window.

1 Initialize the working list.
  - a Sort the $2r+1$ elements in the first window by non-increasing order, and record their indices in array $T$.
  - b Delete those elements that are smaller than their left neighbour.

2 The leftmost element of array $T$ is the index of local maximum in the current window, i.e., $M(r) = f(T(1))$. If $T(1)$ indicates the leftmost element of $\{f(i)\}$ in the current window, delete it. This means that $f(T(1))$ will not be taken into account when the window slides to the right.

3 Slide the current window to the right, and insert the new element $f(k)$ into the proper position from the right end:
  - a If $\exists l, f(T(l-1)) > f(k) \geq f(T(l))$, insert $k$ to the $l$th position of $T$, and delete all elements of $T$ after $l$th.
  - b If $\forall j \in T, f(k) \geq f(j)$, insert $k$ to the first position of $T$, and delete all other elements of $T$.
  - c If $\forall j \in T, f(k) < f(j)$, insert $k$ to the end of $T$.

4 If the current window is not the last one, go to step 2, otherwise the procedure terminates, and the sequence of local maximum is stored as $\{M(j)\}$.

It can be proven that only 3 max/min comparisons are needed for each pixel in 1D extrema filtering, and 6 max/min comparisons are needed for 2D cases [27]. The independence of computational complexity from the size of local window makes the proposed item extraction method applicable to real-time systems.

## 4 Experiments and discussion

The multi-bank processing difficulties mainly result from the variations in text locations and the complexity of their backgrounds as shown in a few examples in Fig. 7. Since each bank owns its specific check design and the location of items, the accuracy of the item extractor is thus essential to the performance of the whole system and is by no means trivial. We carried out both qualitative and quantitative experiments on a training set of 499 real-life gray-level check images scanned at Bell Quebec with a resolution of 300 DPI.

### *4.1 Goal-directed evaluation of extractors*

The item extraction from bankcheck images is a complex document analysis problem which includes many image processing techniques such as enhancement, edge detection, line detection and removal, binarization, and detection of connected components, etc. The selection of appropriate methods for each stage is an important task. Instead of evaluating the quality of extracted items by visual or machine-computed criteria [4,3], an objective evaluation method should be conducted to investigate the effects of different extracting strategies on the subsequent image analysis steps. To evaluate the performance and the effects of extractors on the subsequent recognizers, we followed the goal-directed evaluation used by Trier et al. [23], in which an image understanding module based on the results of the low-level image processing routine in question is used for quantitative evaluation. According to the more popular and more reliable recognition techniques for courtesy amounts, we have chosen the performance of courtesy amounts extractors to evaluate the proposed item extraction method. A courtesy amount recognizer developed by Strathy et al. [1,28] is used. The digit recognizer was trained on about 340,000 digits from the NIST training database plus about 15,000 digits collected by CENPARMI in experimental checks.

Instead of analyzing the performance on recognition of isolated digits as given by most systems in Table 1, we analyze the recognition of the whole string of the courtesy amount, which is the goal of bankcheck image processing. Actually, the relationship between the check recognition rate and isolated digit recognition rate is not as simple as $N$ (number of isolated digits) order of magnitude [7]. The recognition of the whole string of the courtesy amount depends not only on the quality of extracted items, but also on the confidence of segmentation and recognition on noisy and touching digits. In this paper, we focus on the performance of the item extractor and thus leave this topic open.

### *4.2 Experimental results and comparison*

The experiments have been conducted on 499 real-life bankcheck images. To protect personal confidential information, an experimental check is shown in Fig. 8a. We have chosen the size of the horizontal line structuring element $H$ as 30, which is the longest horizontal stroke of a handwritten character. The vertical structuring element $V$ is slightly taller than the line to be extracted. From Figs. 8b and 8c, it can be seen that the horizontal closing operation smears out the horizontal high-frequency parts while the vertical closing smears out the vertical ones. As a result, horizontal lines can be extracted from the gray-level backgrounds. Since the scanning of these check images is restricted to be upright, all baselines of the set are correctly removed. During the restoration of broken strokes, three dynamic kernels described in [29] are chosen. In realizing gray-level mathematical morphology, the fast algorithm of local minimum and maximum filter described in Sect. 3.4.2 is used, in which less than six max/min comparisons were needed for each pixel [27].

To evaluate the performance of different item extraction methods quantitatively, we compared the method proposed by Liu et al. [12] and the method proposed in this paper, which are referred to as *Extractor 1* and *Extractor 2*, respectively. The recognizer used to evaluate the item extractors is also implemented in two stages, in which different strategies were used in splitting or merging the touching and broken characters. In the following comparison, the two recognizers are referred to as *Recognizer 1* and *Recognizer 2*, of which the latter applied more strategies in splitting and merging distorted characters. Comparisons of the performances of *Recognizer 1* and *Recognizer 2* are shown in Tables 2 and 3, and the graphic illustration of relationships between various variables are shown in Figs. 16–18. In Tables 2 and 3, the *Threshold* in the first column stands for the threshold of confidence for outputs. The first row, 'Threshold=0.0' refers to the non-rejection cases. Comparing the two extractors, it is clear that *Extractor 2* has both a higher recognition rate and a higher reliability than *Extractor 1* at the same level of confidence. It is worthwhile noting that 9 checks are rejected by *Extractor 1* due to failure of baseline analysis, and no check is rejected by *Extractor 2* which uses both the baselines and currency symbol as the guide to item extraction. The best performance is given by *Extractor 2* and *Recognizer 2*, by which 78.56% are correctly recognized for the non-rejection case and 51.90% are recognized at the 1% error rate. Compared with *Extractor 1*, the improvements for non-rejection recognition rates are 6% and 12% by *Recognizer 1* and *Recognizer 2*, and 17% and 12% for the 1% error rate case, respectively. The details of the problems solved by the proposed approach are analyzed in the next section.

### *4.3 Qualitative and quantitative analyses*

Although the best recognition rate for courtesy amount and legal amount in the literature is around 70% (Table 1), which was obtained on a test set of binary images or based on a fixed bank reference database, little has been published on the reasons that cause this relatively low rate compared with full-fledged isolated character recognition. According to the visual inspection of misclassified CAs, it was pointed out by several researchers [2–4] that some characters are considerably deformed and unusual characteristics are introduced by personal styles, careless writing, poor quality, unexpected formats, or other noise. The rejections and confusions by the CA recognizer were attributed to some of the following reasons [4]:

- Bad location of CA (e.g., absence or bad detection of baselines in the CA field).

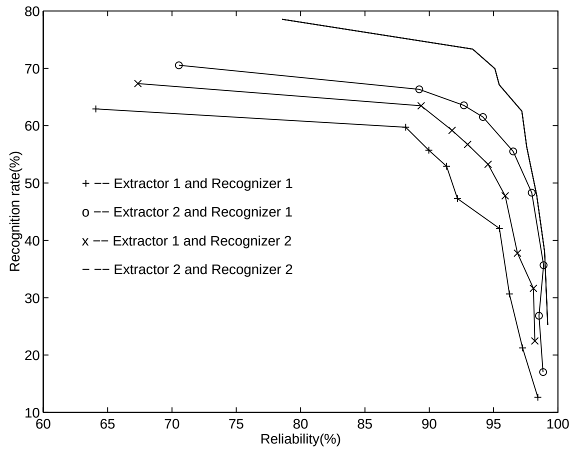



**Fig. 16.** Recognition rate vs. reliability

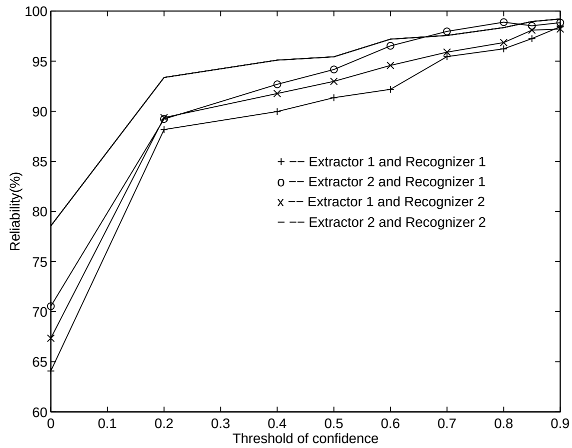



**Fig. 17.** Reliability vs. threshold of confidence

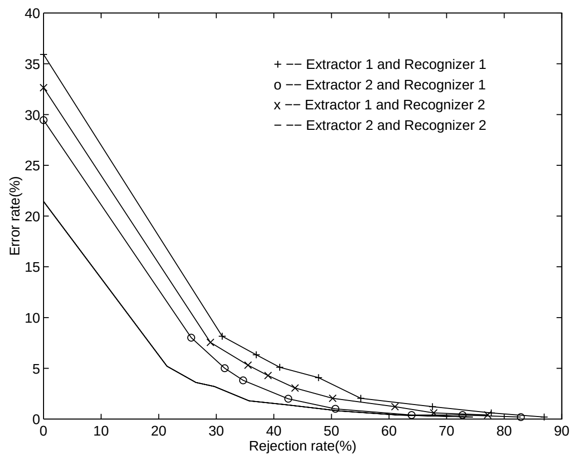



**Fig. 18.** Error rate vs. rejection rate

**Table 2.** Performance at different confidence levels by *Recognizer 1* (whole courtesy amount)

| Extractors | Extractor 1(490/499) | | | | Extractor 2(499/499) | | | |
|---|---|---|---|---|---|---|---|---|
| Threshold | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) |
| 0.0 | 64.08 | 0.00 | 35.92 | 64.08 | 70.54 | 0.00 | 29.46 | 70.54 |
| 0.2 | 60.82 | 31.02 | 8.16 | 88.17 | 66.33 | 25.65 | 8.02 | 89.22 |
| 0.4 | 56.73 | 36.94 | 6.33 | 89.97 | 63.53 | 31.46 | 5.01 | 92.69 |
| 0.6 | 48.16 | 47.76 | 4.08 | 92.19 | 55.51 | 42.48 | 2.00 | 96.52 |
| 0.7 | 42.86 | 55.10 | 2.04 | 95.45 | 48.30 | 50.70 | 1.00 | 97.97 |

**Table 3.** Performance at different confidence levels by *Recognizer 2* (whole courtesy amount)

| Extractors | Extractor 1(490/499) | | | | Extractor 2(499/499) | | | |
|---|---|---|---|---|---|---|---|---|
| Threshold | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) |
| 0.0 | 67.35 | 0.00 | 32.65 | 67.35 | 78.56 | 0.00 | 21.44 | 78.56 |
| 0.2 | 63.47 | 28.98 | 7.55 | 89.37 | 73.35 | 21.44 | 5.21 | 93.37 |
| 0.4 | 59.18 | 35.51 | 5.31 | 91.77 | 69.94 | 26.45 | 3.61 | 95.10 |
| 0.6 | 53.27 | 43.67 | 3.06 | 94.57 | 62.53 | 35.67 | 1.80 | 97.20 |
| 0.75 | 44.08 | 54.08 | 1.84 | 96.00 | 51.90 | 47.09 | 1.00 | 98.11 |

– Bad segmentation of the CA due to problems of background removal.
– Low confidence value caused by segmentation or recognition.
– Bad detection of non-numeral items such as broken strokes or commas.
– Confusion at the numeral recognition stage.

To conduct a quantitative analysis of the experimental results, it is meaningful to classify the problems encountered into finer sub-classes [20], each regarding the problems encountered in each stage of the item extraction procedure. For this reason, we classify the problems into four classes, each of which describes finer details for the types of problems observed: (1) Extraction problems caused by the extractor; (2) Extraction problems caused by unusual writing styles; (3) Recognition problems caused by the recognizer; (4) Recognition problems caused by unusual writing styles. The fine analysis of these reasons is of great importance to the study of future approaches to improving the performance of the general bankcheck processing systems. According to the different reasons for these cases, they are further classified as follows:

1 Extraction problems caused by the extractor:
  a Failure to exclude the "$" symbol or handwritten intruders belonging to other items that intersect the courtesy amounts.
  b Broken strokes or noise caused by the elimination of baselines or bounding rectangular boxes.
  c Broken or touched strokes due to the binarization.
2 Extraction problems caused by unusual writing styles:
  a Failure to extract the courtesy amount due to absent or broken baselines, or the "$" symbol intersected by handwritten strokes or dark background.
  b Failure to extract the cents portion positioned on a different line from the dollar portion.
3 Recognition problems caused by low confidence in isolated or touched characters.
4 Recognition problems caused by unusual writing styles:
  a Failure to recognize the cents portion written in different styles.
  b Failure to recognize scribbled characters.

In order to focus on the discussion of item extraction methods and their respective performance, the recognition problems are not analyzed in this paper. The errors given by *Recognizer 2* in the non-rejection case and 1% error case are analyzed in detail, and the results are shown in Table 4. Compared with the item extraction method previously proposed in [12], problems caused by the extractor have been alleviated. As shown in Table 4, the improvement from *Extractor 1* to *Extractor 2* can be mostly ascribed to the solution of problems caused by case (1), in which problems in case (1)a are mostly solved by bounded connected components grouping, case (1)b by baseline elimination based on gray-level mathematical morphology, and case (1)c by local thresholding, respectively. Cases (2) and (4) are the most difficult problems whose solution calls for the users' cooperation. For problems in case (3), which are caused by low confidence in recognizing isolated or touched characters, a more robust recognizer is required.

Through careful observations, we found that a great deal of the problems were caused by the cents portion, which is often connected with auxiliary elements, such as dash, slash, point, and denominator (either 100 or XX). Since the dollar portion and cents portion share different significance in real applications, it is helpful to give the performance of the courtesy amount by dollar portion recognition. The results evaluated by *Recognizer 1* and *Recognizer 2* are shown in Tables 5 and 6, respectively. The overall performance of courtesy amount recognition given by *Recognizer 2* is presented in Table 7, and the proposed method is summarized in Table 8, using the same notations as in Table 1.

**Table 4.** Quantitative analysis of reasons for errors and rejections

| Extractors | Extractor 1(490/499) | | Extractor 2(499/499) | |
|---|---|---|---|---|
| Problem | Non-rej. errors(%) | 1%-err. rejections(%) | Non-rej. errors(%) | 1%-err. rejections(%) |
| (1)a | 4.49 | 6.53 | 0.40 | 0.60 |
| (1)b | 8.16 | 10.82 | 1.00 | 1.80 |
| (1)c | 2.65 | 2.65 | 1.20 | 1.40 |
| Total(1) | 15.31 | 20.00 | 2.60 | 3.80 |
| (2)a | 1.63 | 1.63 | 0.40 | 0.40 |
| (2)b | 1.02 | 1.02 | 1.40 | 1.60 |
| Total(2) | 2.65 | 2.65 | 1.80 | 2.00 |
| Total(3) | 7.14 | 31.22 | 9.82 | 31.06 |
| (4)a | 3.06 | 5.10 | 4.21 | 5.01 |
| (4)b | 4.49 | 5.92 | 4.21 | 5.21 |
| Total(4) | 7.55 | 11.02 | 8.42 | 10.22 |
| Total | 32.65 | 64.90 | 22.65 | 47.09 |

**Table 5.** Performance at different confidence levels by *Recognizer 1* (cents portion ignored)

| Extractors | Extractor 1(490/499) | | | | Extractor 2(499/490) | | | |
|---|---|---|---|---|---|---|---|---|
| Threshold | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) |
| 0.0 | 73.06 | 0.00 | 26.94 | 73.06 | 77.96 | 0.00 | 22.04 | 77.96 |
| 0.2 | 68.37 | 25.31 | 6.33 | 91.53 | 73.75 | 21.04 | 5.21 | 93.40 |
| 0.4 | 67.14 | 27.76 | 5.10 | 92.94 | 71.94 | 24.85 | 3.21 | 95.73 |
| 0.6 | 61.02 | 35.51 | 3.47 | 94.62 | 67.33 | 30.46 | 2.20 | 96.83 |
| 0.85 | 42.65 | 56.33 | 1.02 | 97.66 | 51.10 | 47.90 | 1.00 | 98.08 |

**Table 6.** Performance at different confidence levels by *Recognizer 2* (cents portion ignored)

| Extractors | Extractor 1(499/499) | | | | Extractor 2(499/499) | | | |
|---|---|---|---|---|---|---|---|---|
| Threshold | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) |
| 0.0 | 72.24 | 0.00 | 27.76 | 72.24 | 81.16 | 0.00 | 18.84 | 81.16 |
| 0.2 | 70.41 | 23.27 | 6.33 | 91.76 | 77.35 | 19.24 | 3.41 | 95.78 |
| 0.4 | 68.57 | 26.73 | 4.69 | 93.59 | 75.55 | 22.04 | 2.40 | 96.92 |
| 0.6 | 65.31 | 31.43 | 3.27 | 95.24 | 70.74 | 27.86 | 1.40 | 98.06 |
| 0.8 | 55.71 | 42.24 | 2.04 | 96.47 | 63.33 | 35.67 | 1.00 | 98.44 |

## 5 Conclusion

The subject of automatic check image processing including layout analysis and understanding has grown rapidly during the past few years. So far, automatic extraction of items from the check images with complicated backgrounds remains as one of the most challenging topics in the field. In this paper, a baseline extraction and elimination method based on gray-level mathematical morphology is proposed. The method extracts long lines using morphological operations instead of parametric estimations. A generic layout-driven item extraction method previously proposed by our group is then combined with a bottom-up analysis, by which over 95% courtesy amounts are extracted more precisely. To separate the handwritten strokes from the background, a local thresholding technique is used. The proposed approach can be applied not only to standard Canadian checks, but also checks in other countries with structures guided by baselines.

To demonstrate the effectiveness of the proposed approach, qualitative as well as quantitative analyses were conducted based on a set of real-life checks with different styles and complicated background designs. A goal-directed evaluation of the extractors is proposed and tested on the extraction of courtesy amounts. Compared with previous works, great improvement is achieved by the baseline elimination, the information restoration by dynamic kernels and the local thresholding techniques. The quantitative study of the reasons for errors and rejections provides possible improvement of future works, and shows that the main difficulties of courtesy amount recognition lies in segmenting touching digits, and overcoming sloppiness and some unusual writing styles.

Since most business forms are characterized by the presence of horizontal and vertical frames that delimit the usable space, the proposed method can be applied generally to locate the frames with a small skew angle. Here we deal with gray-level images in line extraction because binarization may produce noise or lose information around the line regions, which will affect the performance of the recognizers that follow. To carry out the basic morphological operations efficiently, a fast algorithm is used. Extended to morphological operations in arbitrary angles, at the expense of computational complexity, the proposed method is capable of extracting long-thin features from other types of document image.

**Table 7.** Overall performance in extraction of courtesy amount (*Extractor 2*)

| | Whole courtesy amount(499/499) | | | | Dollar portion(499/499) | | | |
|---|---|---|---|---|---|---|---|---|
| Rate | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) | Recog.(%) | Rej.(%) | Err.(%) | Reliab.(%) |
| Non-rej. case | 78.56 | 0.00 | 21.44 | 78.56 | 81.16 | 0.00 | 18.84 | 81.16 |
| 1%-err. case | 51.90 | 47.09 | 1.00 | 98.11 | 63.33 | 35.67 | 1.00 | 98.44 |

**Table 8.** Summary of the proposed method

| Reference | Raw image | item extraction | Binarization | Baseline removal | Test set | Rec.(%) | Rej.(%) | Err.(%) |
|---|---|---|---|---|---|---|---|---|
| Proposed method | $G$ | Top-down & Bottom-up | Local thresholding | Gray-level Math. Morph. | 499 | 78.56<br>51.90 | 0<br>47.09 | 21.44<br>1.00 |

*Acknowledgements.* This research was supported by grants from the Ministry of Education of Quebec, the IRIS Network of Centres of Excellence, the Natural Sciences and Engineering Research Council of Canada, and Fonds pour la Formation de Chercheurs et l'Aide à la Recherche. The authors would also like to thank Nicholas Strathy for introducing the digit recognizers, and Christine Nadal for preparing the check databases.

## References

1. C. Y. Suen, L. Lam, D. Guillevic, N. W. Strathy, M. Cheriet, J. N. Said, R. Fan.: Bank check processing system. Int. J. of Imaging Systems and Technology, 7:392–403, 1996
2. G. Dimauro, S. Impedovo, A. Salzo.: Automatic bankcheck processing: A new engineered system. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):467–504, 1997
3. S. Knerr, V. Anisimov, O. Baret, N. Gorski, D. Price, J. C. Simon.: The A2iA intercheque system: Courtesy amount and legal amount recognition for French checks. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):505–548, 1997
4. L. Heutte, P. Barbosa-Pereira, O. Bougeois, J. V. Moreau, B. Plesis, P. Courtellemont, Y. Lecourtier.: Multi-bank check recognition system: Consideration on the numeral amount recognition module. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):595–618, 1997
5. M. Leroux, E. Lethelier, M. Gilloux, B. Lemarie.: Automatic reading of handwritten amounts on French checks. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):619–638, 1997
6. L. L. Lee, M. G. Lizarraga, N. R. Gomes, A. L. Koerich.: A prototype for Brazilian bankcheck recognition. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):549–570, 1997
7. H. Su, B. Zhao, F. Ma, S. Wang, S. Xia.: A fault-tolerant Chinese check recognition system. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):571–594, 1997
8. G. Dzuba, A. Filatov, D. Gershuny, I.Kil, V. Nikitin.: Check amount recognition based on the cross validation of courtesy and legal amount field. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):639–656, 1997
9. G. Kim, V. Govindaraju.: Bankcheck recognition using cross validation between legal and courtesy amounts. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):657–674, 1997
10. A. Agarwal, L. Granowetter, K. Hussein, A. Gupta.: Detection of courtesy amount block on bank checks. In: Proc. of ICDAR, 2:748–751, 1995
11. M. Okada, M. Shridhar.: Extraction of user entered components from a personal bankcheck using morphological subtraction. Int. J. Pattern Recognition and Artificial Intelligence, 11(5):699–715, 1997
12. K. Liu, C.Y.Suen, M. Cheriet, J. N. Said, C. Nadal, Y.Y. Tang.: Automatic extraction of baselines and data from check images. Int. J. Pattern Recognition and Artificial Intelligence, 11(4):675–697, 1997
13. Yuan Y. Tang, Hong Ma, Jiming Liu, Bingfa Li, Dihua Xi.: Multiresolution analysis in extraction of reference lines from documents with gray level background. IEEE Trans. on Pattern Analysis and Machine Intelligence, 19(8):921–926, 1997
14. M. Cheriet, J. N. Said, C. Y. Suen.: A recursive thresholding technique for image segmentation. IEEE Trans. on Image Processing, 7(6):918–921, 1998
15. J. Serra.: Image Analysis and Mathematical Morphology. London: Academic Press, 1982
16. R. Haralick, S. Sternberg, X. Zhuang.: Image analysis using mathematical morphology. IEEE Trans. on Pattern Analysis and Machine Intelligence, 9(4):532–550, 1987
17. G. Dimauro, S. Impedovo, G. Pirlo, A. Salzo.: Removing underline from handwritten text: An experimental investigation—progress in handwriting recognition. In: A. C. Downton and S. Impedovo (eds.), Progress in Handwriting Recognition, pp. 497–501. World Scientific, 1997
18. D. Guillevic, C. Y. Suen.: Cursive script recognition: A faster reader scheme. In: Proc. of ICDAR, 1:311–314, 1993
19. M. Cheriet, J. N. Said, C. Y. Suen.: A formal model for document processing of business forms. In: Proc. of ICDAR, 1:210–213, 1995
20. J. N. Said.: Automatic Processing of Documents and Bank Cheques. PhD thesis, Concordia Univ., 1998
21. N. Ostu.: A threshold selection method from gray-scale histogram. IEEE Trans. on Syst. Man & Cybern., 8:62–66, 1978
22. J. Bernsen.: Dynamic thresholding of grey-level images. In: Proc. of ICPR, 2:1251–1255, 1986
23. Ø. D. Trier, A. K. Jain.: Goal-directed evaluation of binarization methods. IEEE Trans. on Pattern Analysis and Machine Intelligence, 17(12):1191–1201, 1995
24. Ø. D. Trier, A. K. Jain.: Evaluation of binarization methods for document images. IEEE Trans. on Pattern Analysis and Machine Intelligence, 17(3):312–314, 1995

25. W. Niblack.: An Introduction to Digital Image Processing. Englewood Cliffs, NJ: Prentice Hall, 1986
26. S. D. Yanowitz, A. M. Bruckstein.: A new method for image segmentation. Computer Vision, Graphics and Image Processing, 46(1):82–95, 1989
27. X. Ye, M. Cheriet, C. Y. Suen.: A fast algorithm for extrema filtering in image processing. Technical report, CENPARMI, Concordia Univ., 1998
28. N. W. Strathy, C. Y. Suen, A. Krzyzak.: Segmentation of handwritten digits using contour features. In: Proc. of ICDAR, 1:577–580, 1993
29. J. N. Said, M. Cheriet, C. Y. Suen.: Dynamic morphological processing: A fast method for baseline extraction. In: Proc. of ICPR, 1:8–12, 1996

**Xiangyun Ye** received her Ph.D. degree in Communication and Electronic Systems from Shanghai Jiao Tong University in 1997, M.E. and B.E. degree in Biomedical Engineering from Southeast University in 1994 and 1991, respectively. Since October 1997, Dr. Ye has been with CENPARMI at Concordia University and LIVIA at École de Technologie Supérieure de l'Université du Québec, as a post-doctoral fellow. Her research interests include image processing, pattern recognition, character recognition and document analysis.

**Mohamed Cheriet** received his Ph.D. degree in Computer Science from Université de Paris 6 (France) in 1988. From 1988 to 1990, he worked as a Research Assistant at LAFORIA/CNRS laboratory. He then joined CENPARMI at Concordia University in Montreal, where he worked as a postdoctoral fellow for two years. He was appointed Assistant Professor in 1992 and Associate Professor in 1996 in the department of Automated Production Engineering at the École de Technologie Supérieure de l'Université du Québec in Montreal. His research focuses on image processing, pattern recognition, character recognition, text processing, handwritten documents analysis and recognition, and perception. Dr. Cheriet is a member of IEEE and an active member at CENPARMI. He has published over 50 technical papers in the field.

**Ching Y. Suen** joined the Department of Computer Science of Concordia University, Montreal, in 1972. Presently he is the Director of CENPARMI, the Centre for Pattern Recognition and Machine Intelligence of Concordia. Prof. Suen is the author/editor of 11 books on subjects ranging from computer Vision and Shape recognition, Handwriting Recognition and Expert Systems, to Computational Analysis of Mandarin and Chinese. Dr. Suen is the author of more than 250 papers on these subjects. A fellow of the IEEE, IAPR, and the Royal Society of Canada, Dr. Suen is an Associate Editor of several journals related to Pattern Recognition. He is the Past President of the Canadian Image Processing and Pattern Recognition Society, and the Chinese Language Computer Society. He is Founder or Co-founder of several conferences, including Vision Interface, IWFHR, and ICDAR. Prof. Suen is the recipient of several awards for outstanding contributions to pattern recognition, expert systems, and computational linguistics.

**Ke Liu** received B.S. degree from Nankai University, China in 1982, and M.S. and Ph.D degrees from Nanjing University of Science and Technology (former East China Institute of Technology), China, in 1985 and 1989, respectively. He served as an Assistant Professor and Associate Professor in the Department of Computer Science, Nanjing University of Science and Technology from 1989 to 1993, where he has been a Professor since April 1993. He was a Canada International Fellow at CENPARMI, the Centre for Pattern Recognition and Machine Intelligence of Concordia University of Canada, from October 1993 to October 1995. He has joined CENPARMI as a senior researcher since October 1995 and the Faculty of Engineering and Computer Science of Concordia University as an Adjunct Professor since June 1996. Dr. Liu's interests are pattern recognition, computer vision, neural network, image processing and their applications in handwriting recognition, document image analysis, and so on. He has published over 70 scientific and technical papers in international journals and conferences.