A FAST WAY TO LOCATE THE ITEM REGIONS OF GREY LEVEL IMAGES OF CHEQUES

ZHENG LI

A THESIS
IN
THE DEPARTMENT
OF
COMPUTER SCIENCE

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE
CONCORDIA UNIVERSITY
MONTRÉAL, QUÉBEC, CANADA

APRIL 2001

0-612-59334-7

Abstract

A Fast Way To Locate the Item Regions Of Grey Level Images of Cheques

Zheng Li

In the business transactions of large corporations such as utility companies and banks, many cheques are being processed on a regular basis. Automatic reading of bank cheques is an active topic in Document Analysis. A machine capable of reading bank cheques will have wide applications in banks and those companies where huge quantities of cheques have to be processed. To realize such a system, many image processing, pattern recognition and OCR techniques must be involved.

However, to develop an effective item extraction system is a difficult task, especially when the cheques contain complicated and colourful background pictures. In this thesis, a novel approach is proposed to locate the baselines of the cheque quickly and efficiently from a grey-scale cheque image. Based on that, we can extract the items such as the legal amount, courtesy amount and date respectively via a layout-driven extraction method.

We have developed a system that can extract cheque items effectively and automatically combining all the current methods. The result is quite encouraging and a reliability of 98% has been achieved when tested on IRIS database which contains 600 gray-level cheque images. The performance analysis also indicates that our program is less time-consuming than the existing extraction system.

Acknowledgments

First of all, I would like extend my sincere gratitude to both of my supervisors Dr. Ching Y. Suen and Dr. Ke Liu for their guidance, encouragement, and help throughout my study at Concordia. The time and effort they dedicated to the project and my thesis are crucial and indispensable.

I am grateful to people whose studies have directly contributed to my thesis. Among them are Nicholas Strathy whose image processing code libraries were used in my program, Dr. Yuntao Qian and Jianxiong Dong who both made helpful suggestions to different aspects of this project, Christine Nadal who collected and maintained the cheque image database. Also, I would like to thank William Wong, Michael Assels and Michael Yu for their effort in maintaining a high-quality computing enviroment.

Special thanks should go to my colleagues and fellow graduate students Jie Zhou, Qizhi Xu, Dr. Xiangyun Ye, Boulos Waked, Jun Zhou, Liqiang Zhang, Dr. Xiaoping Chen, Dr. Xuejing Wu and Guiling Guo who have given me their help at different times and shared my emotions.

Last, I am indebted to my parents who brought me up with endless love and unbounded understanding. To Xiaomei, my unique wife, I am so grateful for her effort of always being there for me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Review

Automatic reading of bank cheques has become a very active topic in Document Analysis. A machine capable of reading bank cheques will have wide applications in banks and those companies where huge quantities of cheques have to be processed. So far many research activities on the automatic processing of bank cheques[1, 2, 3] have been reported and concentrated on the challenging academic topics such as the recognition of legal amount [4, 5, 6, 7] (i.e. cursive English or French amount), the recognition of courtesy amount [8, 9] (i.e. handwritten numeral amount), the recognition of date [1], the detection of courtesy amount block on bank cheques [10], and so on. Studies have also been made on real applications such as automatic reading of two handwritten digit amounts and cursive script amounts from French postal cheques [10], automatic reading of the names and addresses of receivers, account numbers, and the amounts of money, etc. from GIRO cheque forms used in Switzerland [3], automatic processing of financial documents [2, 11], and so on. In the research and development of an automatic reading bank cheque reading system, a big cheque image database which contains hundreds or thousands of cheque images is usually used for training and testing. Although the colour image of a cheque contains richer information than that of its grey scale image, it will take much more memory (sometimes also more time) to process a colour cheque image and much more space to store it. Therefore, the main trend of the current research on automatic processing of bank cheques is based on the grey scale or binarized cheque images.

In Canada, for banks and many companies, such as Bell-Quebec Inc. and Hydro-Quebec Inc., etc, a large number of cheques and payment slips(e.g. the cheques for paying credit card charges, telephone bills, and so on) need to be processed every day. When the customers' cheques are processed, the most important pieces of information to be read from these cheques are the account numbers, legal amounts, courtesy amounts and dates.

Figure 1 shows a typical Canadian cheque. In banks and utility companies, the account number in magnetic ink on a cheque can now be automatically read by machine, however, the other items, especially legal amount and date are still processed by operators through human machine interfaces. The overall goal [1, 12] of our research is to develop an automatic system which can read the legal amount, courtesy amount plus the date, and validate the two amounts on a Canadian cheque, so that the processing of the cheques of payment in Canada can be totally automatic.

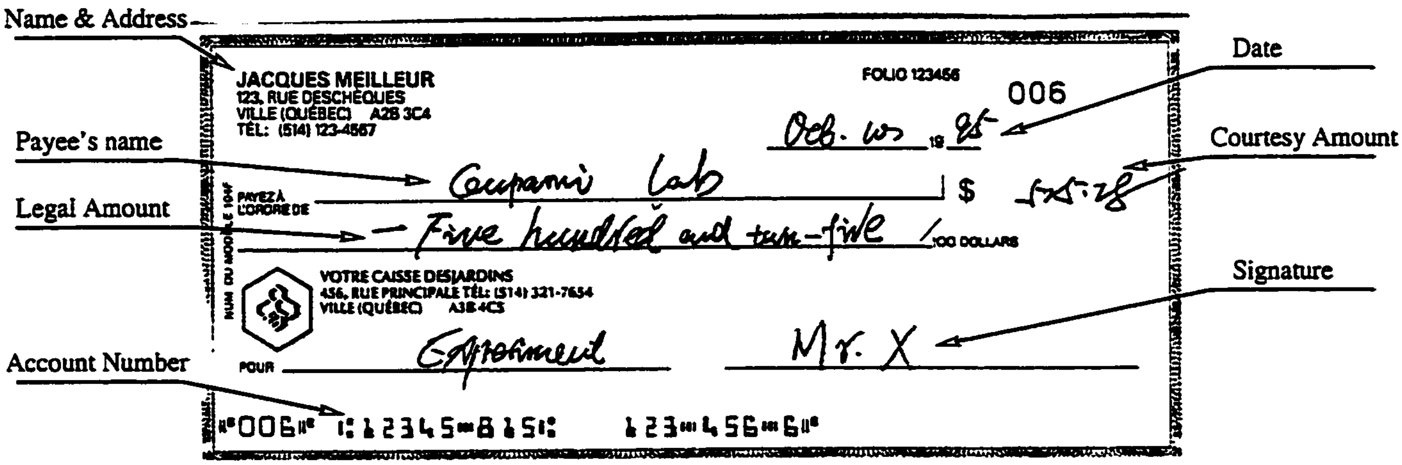


Figure 1: Example of a typical Canadian bank cheque

1.2 Challenge

Apparently, the first and basic step of automatic reading of cheques is item extraction, i.e. to extract the interested data, such as courtesy and legal amounts, date, etc. Only after these items have been correctly extracted, one can start the process of recognition. In general, there are two ways to extract the desired items from a cheque image: (a) first binarize it, then analyse the binarized cheque image and, (b) analyse the original grey scale image directly. It is clear that in the first way, a robust thresholding method is required to binarize the cheque images and to eliminate or reduce the background from them. In our opinion, cheque item extraction methods based on the first way will be useful in those applications where only specific styles of cheques with simple background pictures(see Figure 1) are used or the subregions corresponding to the items (such as courtesy amount, etc) to be extracted from a cheque possess simple background pictures (see Figure 2).

However, if a real bank cheque contains a very complicated background, it is very difficulty to find an effective thresholding method to produce a satisfactory binarized cheque image.

Since Canadian cheques may be printed in different sizes and styles, and each style may possess a variety of colourful pictures in the background(see figure 3), the first way of extracting items described above is not appropriate for our application.

Actually, in most publications, the methods of extracting the interested items from cheques are based on this way. Furthermore, few published papers provide the detailed methods of extracting of legal amounts or courtesy amounts from cheques. Ref. [13] proposed a method of detecting courtesy amount block on bank cheques based on binarized cheque images, but the extraction of legal amount or dates is not considered, whereas our study shows that for many Canadian cheques, the complicated backgrounds are frequently in the sub-regions corresponding to legal amounts, payees' information, and dates, etc. Therefore, in this thesis, a novel and generic approach to extract legal and courtesy amounts and dates from Canadian cheques is presented by analysing grey scale cheque images directly.

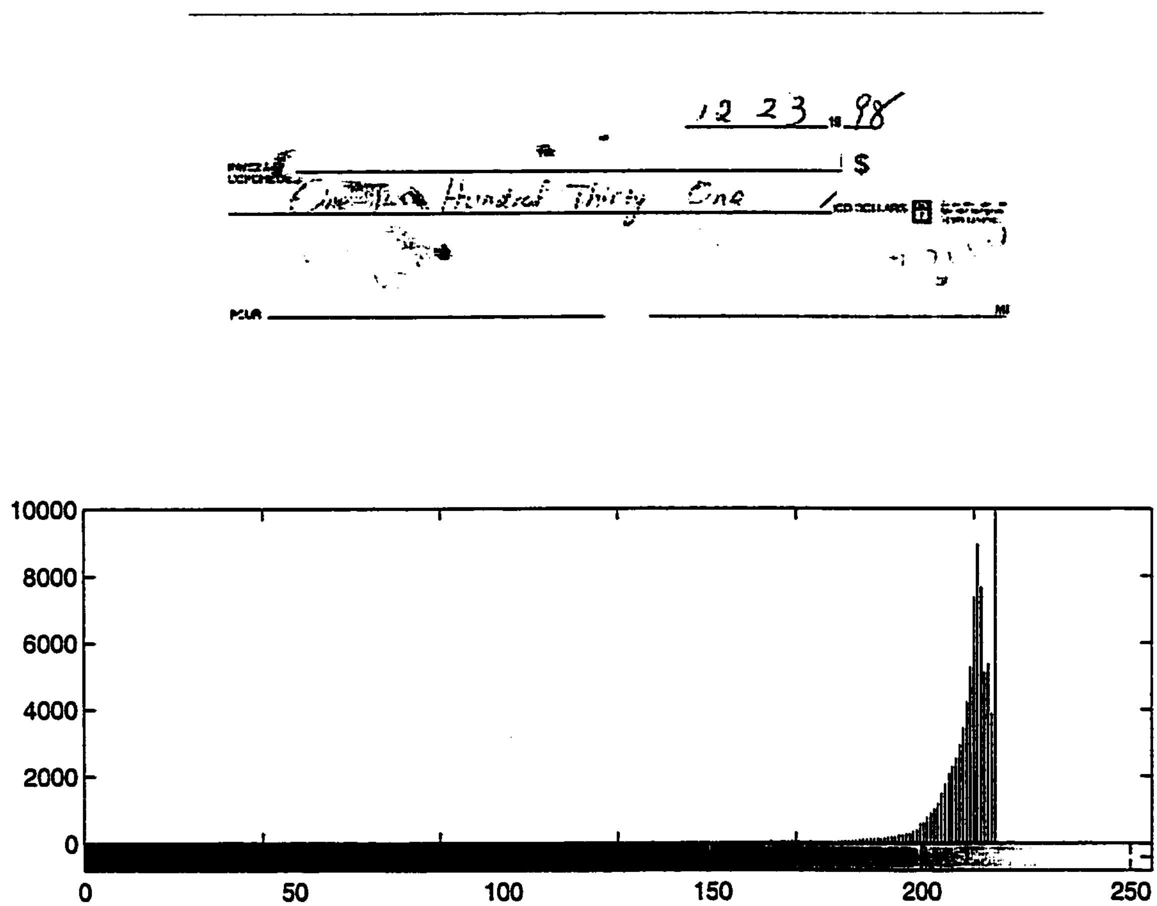


Figure 2: Cheque image considered as having a simple homogeneous background

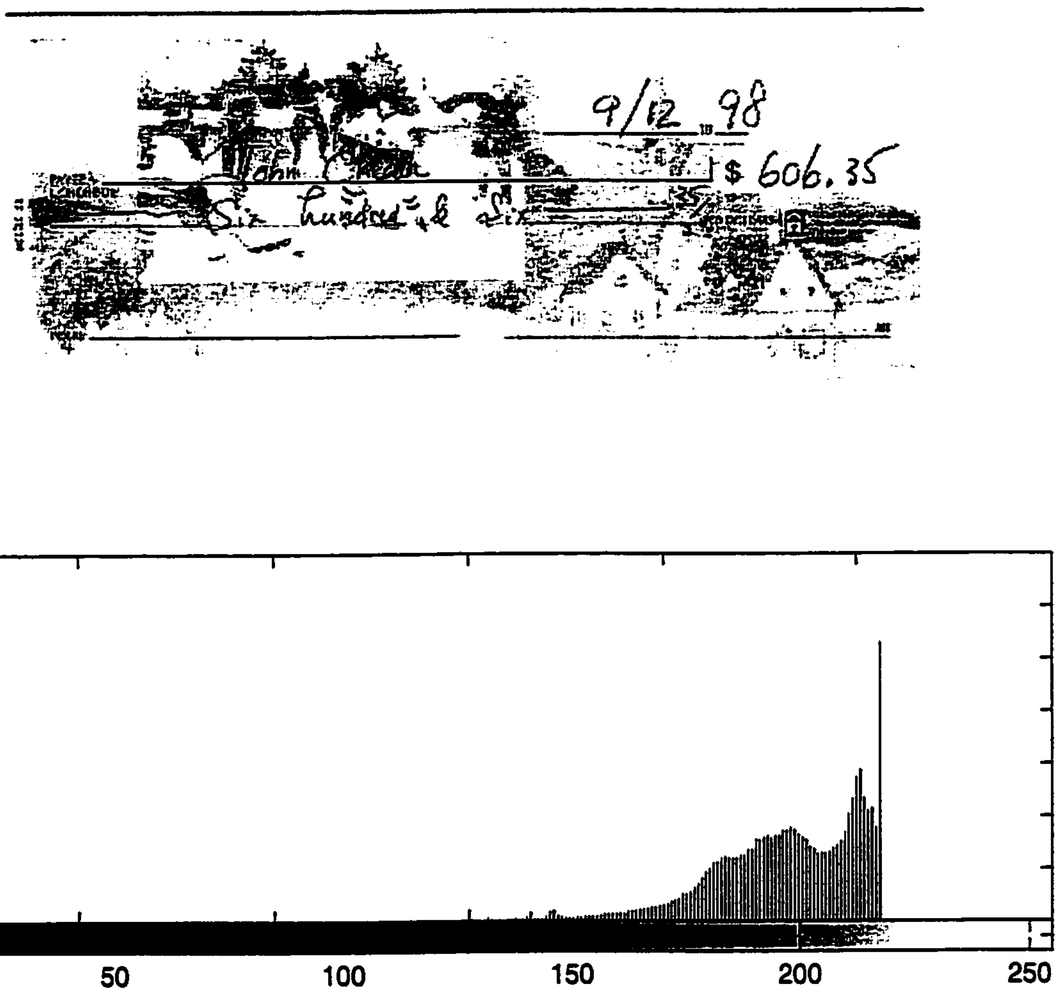


Figure 3: Cheque image considered as having a complex background

1.3 Novelty of the New Approach

The novelty of this new approach is to eliminate the background, first, by using recursive dynamic thresholding technique that could be used globally or locally on a given cheque image. As a second step, baselines that could intersect the handwritten data are recognised by introducing a fast method according to the features of normal angle of those baselines' points and then removed with the challenge of minimizing the distortion of the extracted items. Two methods are proposed to tackle this difficulty. The first method acts as a detector of the handwritten data that intersect with the baselines that should be eliminated and uses morphological and top-logical processing to identify and fill the gaps resulted after eliminating the detected baselines. The second method proposed a new dynamic morphological processing technique which acts as a detector and a preserver of the handwritten data that intersect with the baselines that should be eliminated. The second method highly advances the efficiency of item extraction by more than 80% and enhances the quality of the extracted items when combined with local processing techniques(see Figure 4).

1.4 Thesis Organization

The remainder of the thesis is organized into the following chapters:

Chapter 2 will briefly introduce an existing similar cheque image processing system from CENPARMI lab and point out its bottleneck and relationship to my work.

Chapter 3 will mainly deal with baseline detection by introducing a fast and robust method by taking advantage of the gradient directions of the edge points .

Chapter 4 presents a new generic layout-driven method for the extraction of items from grey scale cheques. Besides, the morphological processing of gray-scale image will be presented to point out the important usage of morphological processing in our approach. Morever, the concept of topological processing will be presented as a solution to the problem left out by morphological processing.

Chapter 5 will present an analysis of the experimental results in order to evaluate the efficiency and reliability of the developed system. A comparative test between

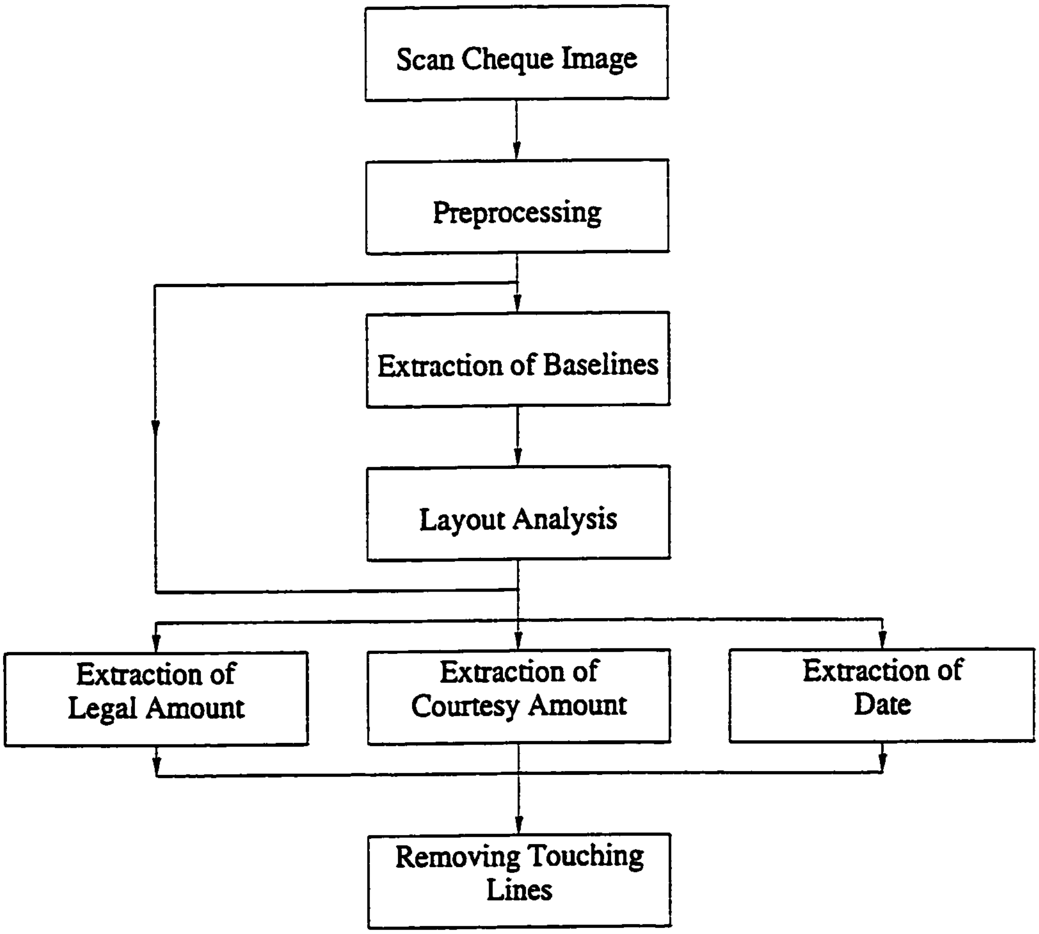


Figure 4: Diagram of cheque item extraction system

our approach and another item extraction method will be given.

Finally, chapter 6 will list the major contribution of this thesis and the future research will be presented as well.

Figure 5 illustrates the organization of this thesis.

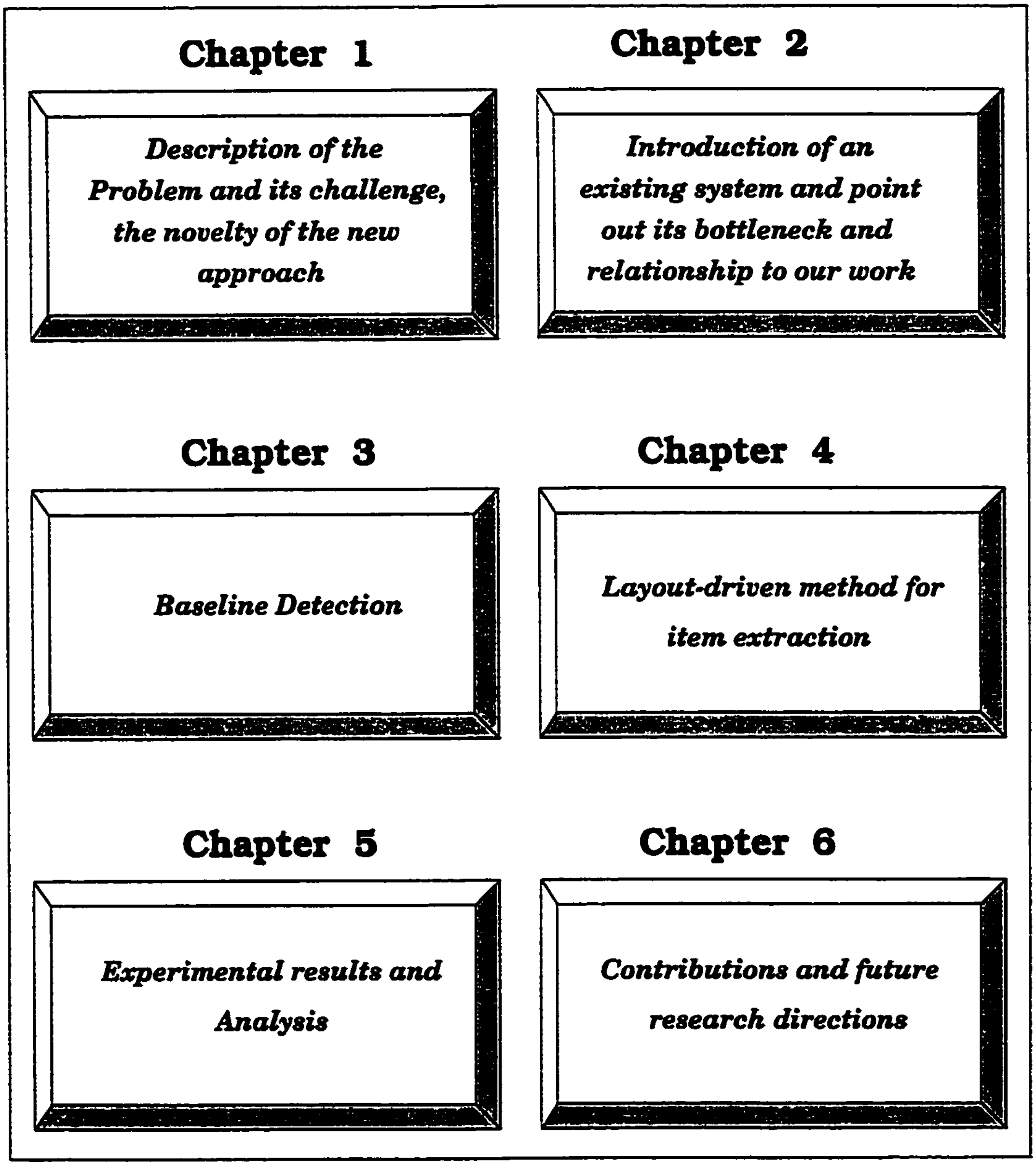


Figure 5: Organization of the Thesis

Chapter 2

Similar Work and Its Relationship to This Work

2.1 Introduction

As a branch of pattern recognition and artificial intelligence, document analysis and recognition has grown rapidly during the past few years. Numerous techniques and methods have been explored and some document processing systems have been proposed [14, 15]. However, most of the studies are concentrated on non-financial types of documents such as newspapers, magazines, technical reports, bibliographical documents, envelopes, and so on. Only a few are related to the processing of business documents in spite of the fact that millions of business transactions take place every day. Associated with them are documents such as bank cheques, payment slips and bills which have to be processed. A great deal of time, effort and money will be saved if such documents can be entered into the computer and processed automatically. However, not withstanding major advances in computer technology and intensive research efforts, the degree of automation in processing such documents is still very limited and a lot of manual labour is currently spent in this area.

The above situation indicates that any system which can speed up the processing of such documents will make a significant contribution. In [2], a fixed document processing subsystem and a flexible document processing subsystem are separately used to process different types of financial documents according to their structural complexity. The purpose of the fixed subsystem is to process documents that have common properties in their geometrical and their logical structures, such as bank cheques. The purpose of the flexible subsystem is to process documents that do not necessarily have common properties in their geometrical or logical structures, such as different types of income tax forms. Both subsystems are robust and efficient in describing the structural properties of a binarized document. However, to improve the quality of the information to be extracted from a given document by perserveing its topological properties and to increase the recognition rate of the extracted information from the documents, binarized documents should be replaced by grey-scale documents.

2.2 An Existing System

Here I want to briefly introduce an existing system **APES** (Automatic Payment Extraction System) for cheques from CENPARMI lab. This system mainly deals with grey level cheque image processing, it consists of four steps: preprocessing, baseline detection, extraction of courtesy amount, legal amount and date, and restoration of lost information.

2.2.1 preprocessing

In **APES**, preprocessing is completed through two steps: first an edge detector is applied to the cheque image to calculate its gradient images (both edge strength and normal direction images), then thresholding techniques is used to select the candidate edge points and to calculate the modified normal directions of edges. The well-known Sobel operator is employed in the system because it is simple and effective in the presence of noise, and can provide gradient directions of edge points. In order to suppress the noise in an input cheque image, a smoothing operator [16] is also applied before edge detection.

The gradient direction angle (degree) at a pixel calculated by Sobel operator fall within the range of [-90, 90], and it is extended to [0, 360] to incorporate the constraint of edge directions in the subsequent processing of the system. Furthermore, the normal directions of local edges have to point at those of their corresponding grey regions with lower grey values.

2.2.2 Extraction of baselines

The least square fitting [17] and Hough transform [18, 19] are two important techniques employed in **APES** which have been extensively used for fitting straight lines from edge images.

First the least square fitting technique applied to the line segments extracted from the edge image to determine an initial estimation of the parameters of the straight line segments and the direction of baselines. Then, a method based on Hough transform technique is proposed to refine the calculated parameters and the direction angle of baselines in a global manner. The method can be summarized as follows:

1. Extracting line segments.
2. Fitting straight line segments based on the least square fitting criterion.
3. Calculating the average directional angle.
4. Refining the directional angle(based on Hough transform).
5. Connecting straight line segments.

2.2.3 Cheque items Extraction

Having located and identified the location, size and orientation of each straight line segment in the cheque image, layout analysis in **APES** is facilitated by having apriori knowledge about the layout structure of the real life cheque images.

After performing the layout analysis of the cheque image, the bounding and searching regions of each item to be extracted are determined. Then the **APES** system will consider these regions for each item to estimate the grey level distributions for each item, this helps to distinguish between the grey level distributions of the handwritten data and other pixels, thresholding is performed in order to segment the handwritten data from the background. Extraction of the items is done by finding the connected components of which the grey values of their connected points are lower than a threshold T within a restricted region.

2.2.4 Restoring Lost Information

The connected components extracted from a cheque image may contain both strokes and lines which either touch or cross each other, or occasionally the bounding box lines of the courtesy amount. Therefore, a post-processing is needed to remove such lines which do not belong to the strokes of the legal and courtesy amounts, and date from the connected components. The methods employed in **APES** sytem are summarized as follows:

1. Elimination of baseline.

2. Morphological closing operation. The purpose of using this operation here is to increase the difference in the intensity between a baseline and the handwritten data that intersects with that baseline.

3. Topological processing. Sometimes, the intensity of the baselines maybe very close to that of the handwritten data, topological operation is applied to fill the gaps as much as possible.

2.3 The bottleneck of the system

Experiments indicate that this system (**APES**) works quite well except for the time consumed. After careful trace and analysis inside the system, we found that the bottleneck occurs at the stage of preprocessing, particularly for the detection and location of the baseline.

So we proposed a new method for fast baseline location to tackle this bottleneck in order to reduce the overall time cost.

Chapter 3

A Fast Way of Baseline Detection

3.1 Description of Cheques

For a standard Canadian cheque, the legal amount is always written above a baseline which is longer than half the height of the cheque. This baseline is called the "legal amount baseline". Above this is another baseline which is also longer than half the height of the cheque for filling in the name of the payee. This baseline is called the " payee baseline". The date is written above two short baselines at the same height defined as the "date baselines"(sometimes date is written above only one short baseline, in such a case, we assume that two baselines coincide). The date baselines are located above the payee baseline on the right hand side of the cheque. The courtesy amount is also located on the right hand side of the cheque between the date and legal amount baselines. For some cheques, there may be a short baseline on the right side of payee baseline and both baselines have the same height. In such cases, the courtesy amount is written above this line which is called the courtesy baseline. Therefore, a cheque will be described by the legal amount baseline, payee baseline, date baselines, and courtesy baseline.

3.2 Image Enhancement

Now, given an image $f(x, y)$, a smoothed image f_{filt} is generated whose intensity at every pixel (i, j) is obtained by averaging the intensity values of the pixels of f contained in a 3 x 3 neighbourhood of (i, j). Formally:

$$f_{filt}(i, j) \;=\; \frac{1}{K} \sum_{(n,m)\in s} f(n, m), \forall(i, j) \in W \times H \qquad (1)$$

where K is the total number of points in S which is assumed as a 3 x 3 neighbourhood, and W and H are respectively the width and height of the image f.

Such smoothing is required in order to produce a better histogram that will be used to generate more accurate thresholding values. Figure 6 illustrates an example.

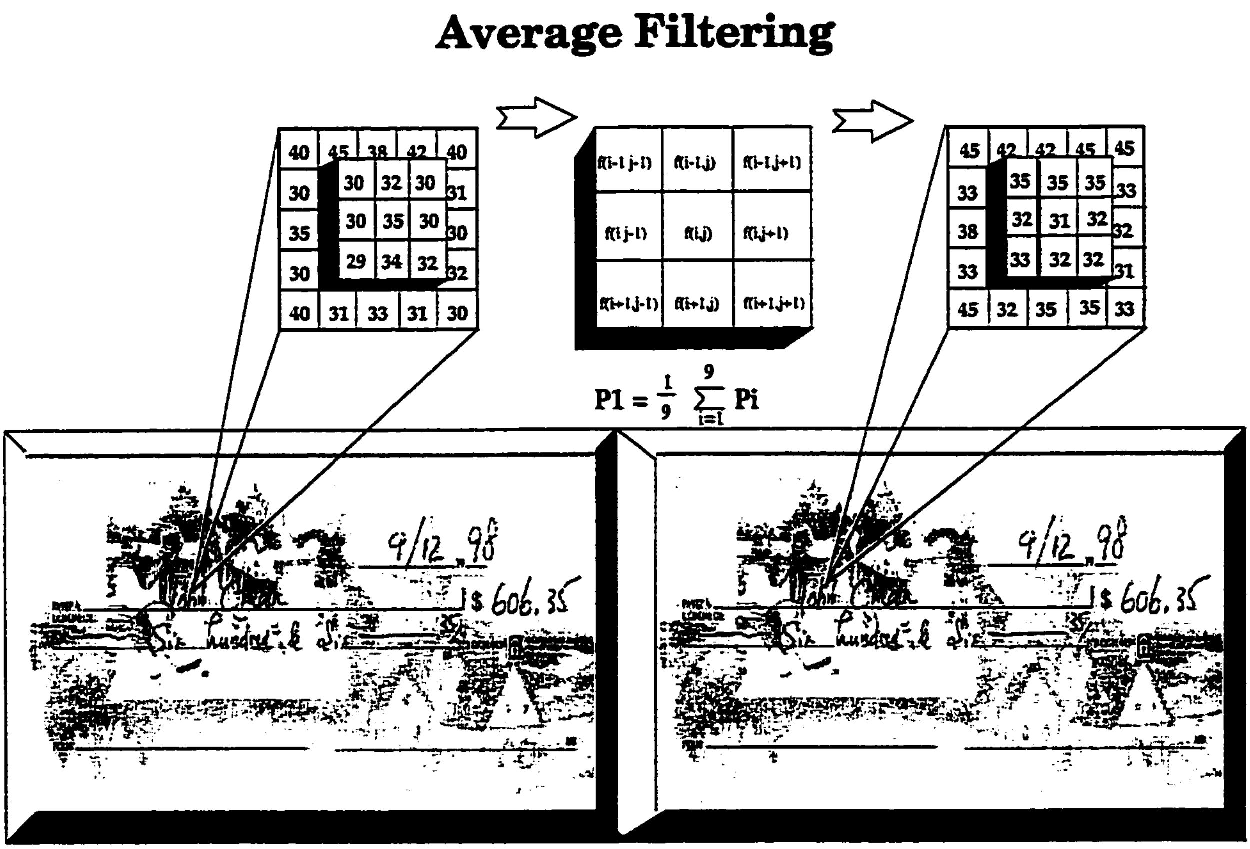


Figure 6: Image enhancement using a 3X3 window

3.3 Background Elimination

3.3.1 Choosing a thresholding method

The key and difficult problem in using the local edge detectors (including Sobel operator) is to determine a threshold which decides between edge and non-edge pixels in the image. A variety of methods [19-25] have been proposed to tackle this problem, which can be divided into global and local thresholding techniques, respectively. In the global thresholding technique, a threshold is determined by analyzing the edge detector output histogram based on statistic models or the criteria to optimize various edge attributes such as location and continuity [24,25], etc, and is applicable to the pixels of the whole image. In local thresholding technique, a threshold is adapted for a local region, or even for a pixel.

Choosing a thresholding method depends on the application. When a cheque image has a very complicated background picture, different regions in the edge image may possess different statistical properties. In such a case, global thresholding may produce thick edges in one region and thin or broken edges in another. Therefore, a local thresholding technique is used in our system.

3.3.2 Otsu's Approach (Clustering)

Let N be the set of natural numbers, (i, j) be the spatial coordinates of a digitized image, and $G = 0, 1, ..., l-1$ be a set of positive integers representing the gray-levels[1]. Then, an image function can be defined as the mapping

$$f : N \times N :\rightarrow G \tag{2}$$

where $f(i, j)$ is the gray-level of a pixel whose coordinates are (i, j), $0 \leq i \leq h$ and $0 \leq j \leq w$, and h and w are the height and width of a digitized image f respectively. Also, by convention, the gray-level 0 is the darkest and the gray-level $l-1$ is the brightest.

[1] In our model we assume $l = 256$.

Let $t \in G$ be a threshold and $B = \{b_0, b_1\}$ be a pair of binary gray-levels such that $b_0, b_1 \in G$. The result of thresholding an image $f(x, y)$ at gray-level t is a binary image function

$$f_B : N \times N :\rightarrow B, \tag{3}$$

such that

$$f_B(i,j) = \begin{cases} b_0 & \text{if } f(i,j) < t. \\ b_1 & \text{if } f(i,j) \geq t. \end{cases} \tag{4}$$

In general a thresholding method is one that determines the optimal value t^* of t based on a certain criterion.

Otsu's method, as proposed in [20], is based on discriminant analysis. The threshold operation is regarded as the partitioning of the pixels of an image into two classes C_0 and C_1 (e.g., objects and background) at gray-level t, i.e., $C_0 = \{0, 1, ..., t\}$ and $C1 = \{t+1, t+2, ..., l-1\}$.

In order to evaluate the "goodness" of the threshold (at level k) the discriminant criterion measures (or measures of class separability) used in the discriminant analysis [21] and defined in Equation 5. In this case, the problem is reduced to an optimization problem to search for a threshold t that maximizes one of the object functions (the criterion measures) of Equation 5.

This standpoint is motivated by a conjecture that well thresholded classes would be separated in gray-levels, and conversely, a threshold giving the best separation of classes in gray-levels would be the best threshold.

As stated in [20], let σ_W^2, σ_B^2, and σ_T^2 be the within-class variance, between-class variance, and the total variance, respectively. An optimal threshold can be determined by maximizing one of the following (equivalent) criterion functions with respect to t :

$$\lambda = \frac{\sigma_B^2}{\sigma_W^2}, k = \frac{\sigma_T^2}{\sigma_W^2}, \eta = \frac{\sigma_B^2}{\sigma_W^2}. \tag{5}$$

These discriminant criterion maximizing λ, k and η, respectively for t are, however, equivalent to one another: e.g., $k = \eta + 1$ and $\eta = \lambda/(\lambda + 1)$ in terms of λ, because the following basic relation always holds:

$$\sigma_W^2 + \sigma_B^2 = \sigma_T^2. \tag{6}$$

It is noticed that σ_W^2 and σ_B^2 are functions of threshold level t, but σ_T^2 is independent of t. It is also noted that σ_W^2 is based on the second-order statistics (class variance), while σ_B^2 is based on the first-order statistics (class means). Therefore, η is the simplest measure with respect to t. Thus η must be adopted as the criterion measure to evaluate the "goodness" (or separability) of the threshold at level t. Thus the optimal threshold t^* is defined as:

$$t^* = Arg \max_{t \in G} \eta, \tag{7}$$

where

$$\sigma_T^2 = \sum_{i=0}^{l-1} (i - \mu T)^2 P_i, \qquad \mu T = \sum_{i=0}^{l-1} i P_i, \tag{8}$$

$$\sigma_B^2 = \omega_0 \omega_1 (\mu_1 \mu_0)^2, \quad \omega_0 = \sum_{i=0}^{t} P_i, \quad \omega_1 = 1 - \omega_0 \tag{9}$$

$$\mu_1 = \frac{\mu_T - \mu_t}{1 - \mu_0}, \quad \mu_0 = \frac{\mu_t}{\omega_0}, \quad \mu_t = \sum_{i=0}^{t} i P_i, \tag{10}$$

n_i is the number of pixels with gray-level i and n is the total number of pixels in a given image defined as

$$n = \sum_{i=0}^{l-1} n_i. \tag{11}$$

P_i is the probability of occurrence of gray-level i defined as

$$P_i = \frac{n_i}{n}. \tag{12}$$

In digital images, the uniformity of objects plays a significant role in separating objects from the background. In fact Otsu's method in thresholding gray-level images is good on the basis of uniformity measure between the two classes C_0 and C_1 that should be thresholded.

3.3.3 Analysis of Further Aspects

Otsu's method as proposed affords further means to analyze additional aspects other than the selection of the optimal threshold for a given image.

For a selected threshold t^* of a given image, the class probabilities ω_0 and ω_1 indicate the portions of the areas occupied by the classes C_0 and C_1 respectively in the thresholded image. The class means μ_0 and μ_1 serve as estimates of the mean levels of the classes in the original gray-level image.

The maximum value of η, can be used as a measure to evaluate the separability of classes C_0 and C_1 in the original image or the binary of the histogram. This is a very significant measure because it is invariant under affine transformations of the gray-level scale. It is uniquely determined within the range

$$0 \leq \eta \leq 1. \tag{13}$$

The lower bound (zero) is obtained when, and only when, a given image has a single constant gray-level, and the upper bound (unity) is obtained when, and only when, two-valued images are given. This property is an important criterion that we will use in extending Otsu's approach.

3.3.4 Recursive Thresholding

Now, given f_{filt}, thresholding as in [20] is applied recursively. Initially, a gray-level histogram is evaluated for the entire image F_{filt}. From this histogram, the method [20] automatically calculates the threshold THR_1 and the separability factor SP_1 for the entire image. SP_1 (η^*) is a number ranging from 0 to 1, it indicates the likelihood of separating the class that has the lowest intensity in the image from the other classes in the same image. If the separability factor is less than S ($S = 95\%$ which is determined after training), which means that the two classes or objects to be separated are not 95% (or more) the same, the threshold value THR_1 will be used to separate the two classes or objects. This results in a new image f_1 from which the object having the highest pixel values (denoted by C_0) has been segmented from the image f_{filt}.

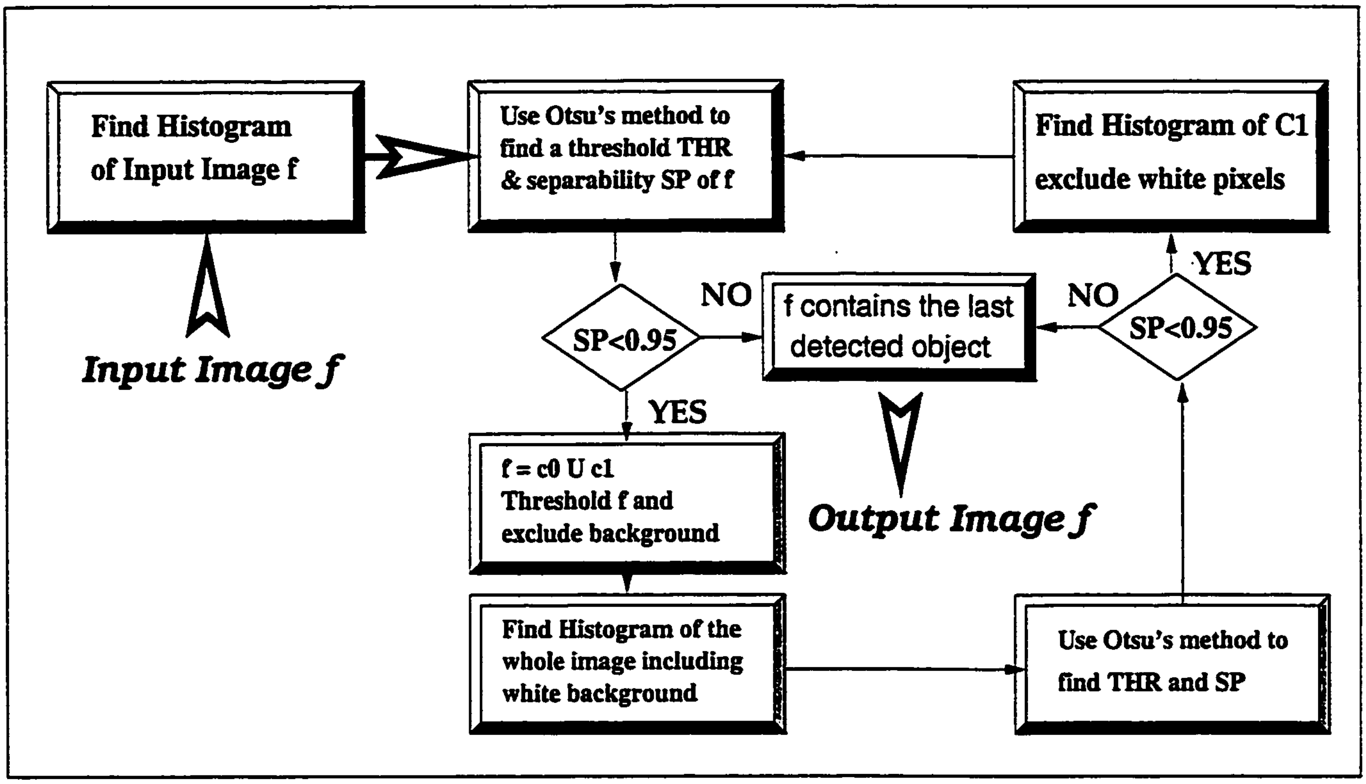


Figure 7: Diagram of Recursive Thresholding Technique

At recursion t ($t \geq 1$), we will determine the gray-level histogram of the whole image including the white pixels of the background and use [20] to calculate the separability factor SP_t and the threshold value THR_t. If SP_t exceeds 95% the process will terminate since the image has only one object remaining against a white background. However, if SP_t does not exceed 95%, the recursive process is repeated to threshold an object C_t. Recursion is repeated p times until no more objects can be segmented. As a result, we obtain:

$$f_{filt} = C_0 \cup C_1 \cup \ldots \cup f_p, \tag{14}$$

where $f_p = f_{thr}$ is the object that remains after segmenting the different objects. Figure 7 presents the diagram.

Baselines correspond to pre-printed lines belonging to the geometrical structure of

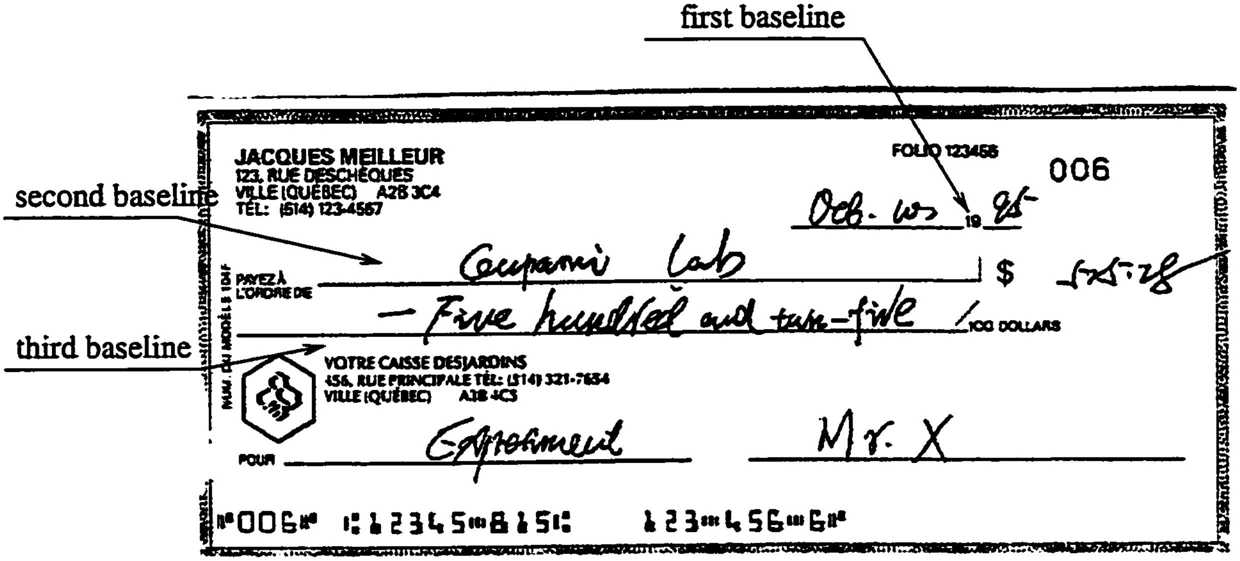


Figure 8: Baselines need to be detected and removed

a financial document. They occur in bank cheques and other types of documents and serve as guidelines to help the writer or the typist to fill data in the proper locations.

After carefully analyzing the geometrical structure of the Canadian bank cheques, we found that they have a common general layout. By scanning any bank cheque from top to bottom, several knowledge rules can be derived, e.g.

- All baselines of interest extend into the area of the rightmost 1/4 of cheques.
- Although space between baselines as well as the position and style of pre-printed data vary in different bank cheques, all baselines are always located in the following order:
    1. The first baseline is for the date
    2. The second baseline is for the courtesy amount.
    3. The third baseline is for the legal amount.
    4. The last baseline is for the signature.

Figure 8 indicates all the baselines we are interested.

3.4 Sobel Operator

It is widely accepted that one of the first stages in machine vision is to extract the primitive constructs of the image. One important primitive construct here is an edge. An accepted view is to define an edge as a point [22] in the image where there is a coherent local change in image luminance or intensity. Several techniques have been developed for detecting edges. Edge operators can be broadly classified into three major groups:

- enhancement/thresholding operators;
- edge-fitting operators; and
- zero crossings of second derivative operators.

The Sobel operator performs a 2-D spatial gradient measurement on an image and so emphasises regions of high spatial frequency that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input gray scale image.

In theory at least, the operator consists of a pair of 3 x 3 convolution kernels as shown in Figure 9. One kernel is simply the other rotated by 90°.

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation(call these G_x and G_y). These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given by:

$$|G| = \sqrt{Gx^2 + Gy^2} \tag{15}$$

Typically, an approximate magnitude is computed using:

$$|G| = |Gx| + |Gy| \tag{16}$$

Sobel Operator

-1	0	1
-2	0	2
-1	0	1

Sobel X Operator

-1	-2	-1
0	0	0
1	2	1

Sobel Y Operator

Figure 9: Sobel Operator

which is much faster to compute. The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by:

$$\theta = \arctan(Gy/Gx) \tag{17}$$

In this case, orientation 0 is taken to mean that the direction of maximum contrast from black to white runs from left to right on the image, and other angles are measured anti-clockwise from this. Often, this absolute magnitude is the only output the user sees — the two components of the gradient are conveniently computed and added in a single pass over the input image using the pseudo-convolution operator. Using this kernel the approximate magnitude is given by:

$$\begin{aligned} |G| &= |(P_1 + 2 \times P_2 + P_3) - (P_7 + 2 \times P_8 + P_9)| \\ &\quad + |(P_3 + 2 \times P_6 + P_9) - (P_1 + 2 \times P_4 + P_7)| \end{aligned} \tag{18}$$

3.5 Fast Detection of baselines

Baseline detecting is a very important process, once these baselines are detected then we can locate those zones which contain the items of interest.

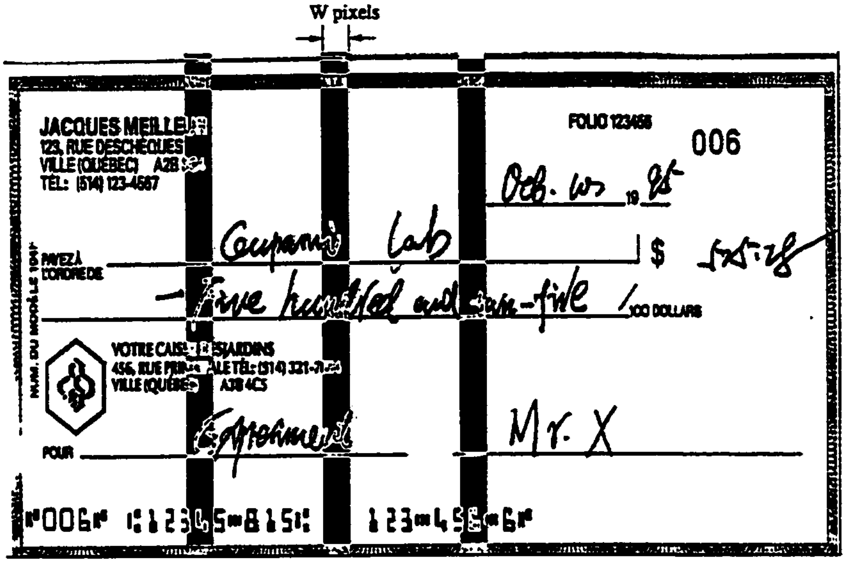


Figure 10: Define N lines crossing the cheque vertically and apply sobel edge detector along these rectangles of W pixels wide

Most existing methods apply Sobel operator or other edge detectors to the entire image of the cheque, detecting all the straight lines at first, then try to locate the baselines we need. It is easy to observe that this kind of algorithm is very time consuming and greatly affects the performance of the cheque processing system. In our approach, only useful straight lines (baselines) are extracted, it consists of three major stages: defining vertical zones, detecting candidate edge points along these vertical lines, and finally applying normal constraints.

First, we define N zones which cross the whole cheque vertically, and the width of each line is W pixels(see Figure 10).

Then, a Sobel edge detector is used to detect the candidate edge points along these N thin regions with W pixels wide, after which the gradient magnitude $D(i,j)$ and gradient direction maps $\theta(i,j)$ of each thin zone f_m are produced, where $0 \leq i \leq H$ and $0 \leq j \leq W$, as:

$$D(i,j) = \sqrt{(\Delta_x f_m(i,j))^2 + (\Delta_y f_m(i,j))^2} \tag{19}$$

$$\theta(i,j) = \tan^{-1}(\Delta_y f_m(i,j))/(\Delta_x f_m(i,j)) \tag{20}$$

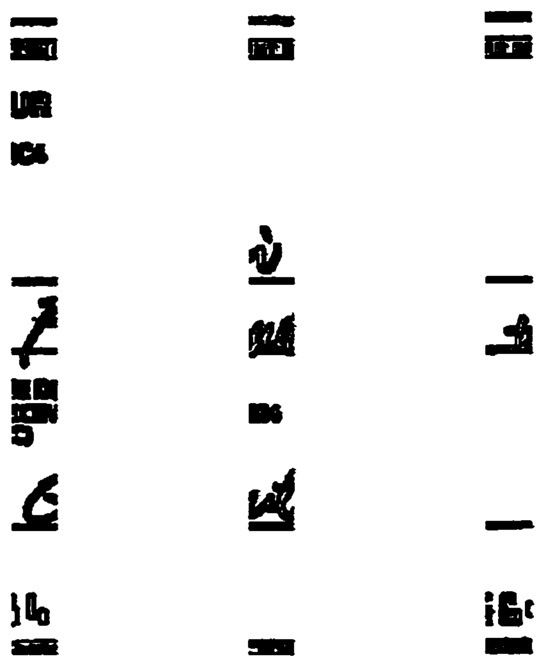


Figure 11: N lines extracted from the original image after Sobel operation

where

$$\begin{aligned}\Delta_x f_m(i,j) &= f_m(i+1,j+1)+2f_m(i+1,j)+f_m(i+1,j-1) \\ &\quad -f_m(i-1,j+1)-2f_m(i-1,j)-f_m(i-1,j-1)\end{aligned} \tag{21}$$

and

$$\begin{aligned}\Delta_y f_m(i,j) &= f_m(i-1,j+1)+2f_m(i,j+1)+f_m(i+1,j+1) \\ &\quad -f_m(i-1,j-1)-2f_m(i,j-1)-f_m(i+1,j-1)\end{aligned} \tag{22}$$

This technique is applied to the image f_m to produce the image f_{edge} that is used later for further processing. Figure 11 illustrates a sample process. After that, we presented a local adaptive thresholding technique to the gradient magnitude map to obtain the final edge points.

3.5.1 Adaptive Thresholding

In many cases, the background gray level is not constant, and the contrast of objects varies within the image. In such cases, a threshold that works well in one area of the image might work poorly in other areas. Adaptive thresholding changes the threshold dynamically over the image. This more sophisticated version of thresholding can accommodate changing lighting conditions in the image, e.g. those occurring as a result of a strong illumination gradient or shadows.

Adaptive thresholding typically takes a gray-scale or colour image as input and, in the simplest implementation, outputs a binary image representing the segmentation. For each pixel in the image, a threshold has to be calculated. If the pixel value is below the threshold it is set to the background value, otherwise it assumes the foreground value.

There are two main approaches to finding the threshold: (i) the *Chow* and *Kanenko* approach, and (ii) local thresholding. The assumption behind both methods is that smaller image regions are more likely to have approximately uniform illumination, thus being more suitable for thresholding. Chow and Kanenko divide an image into an array of overlapping sub images and then find the optimum threshold for each sub image by investigating its histogram. The threshold for each single pixel is found by interpolating the results of the sub images. The drawback of this method is that it is computational expensive and, therefore, is not appropriate for real-time applications.

An alternative approach to finding the local threshold is to statistically examine the intensity values of the local neighbourhood of each pixel. The statistic which is most appropriate depends largely on the input image. Simple and fast functions include the *mean* of the *local* intensity distribution,

$$T = mean$$

the *median* value,

$$T = median$$

or the mean of the minimum and maximum values,

$$T = \frac{\max - \min}{2}$$

The size of the neighbourhood has to be large enough to cover sufficient foreground and background pixels, otherwise a poor threshold is chosen. On the other hand,

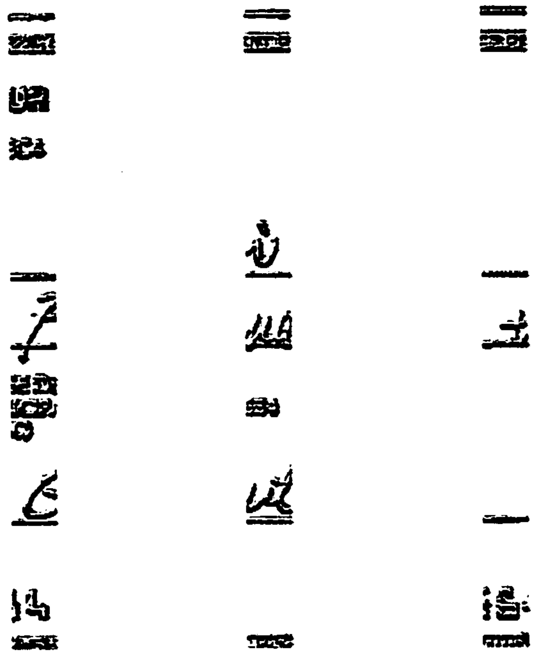


Figure 12: Image after further processing

choosing regions which are too large can violate the assumption of approximately uniform illumination. This method is less computationally intensive than the Chow and Kanenko approach and produces good results for some applications. A task well suited to local adaptive thresholding is in segmenting text from the gray-level images of cheque. A set of all 8 connected edge points forms a connected component, and line segments are extracted as connected components that satisfy conditions both on the length and the gradient directions of the connected edge points. Figure 12 illustrates the image of cheque after the local adaptive thresholding process.

3.5.2 Logical Operation

After all the prior procedures including Sobel operator and local adaptive thresholding, we now have got all the candidate edge points as Figure 12. Then, a logical "AND" operation will be applied among these N zones W pixels wide in order to eliminate those edge points which do not belong to the baselines. After "AND" logical operation, most irrelevant edge points are removed but some edge points of a stroke may still remain as seen in Figure 13, so how to remove these noises and get the right baselines will be a problem.

Figure 13: Image after logical 'AND' processing

3.5.3 Normal Constraint

Suppose the width of the lines to be detected is nearly the same, and their edges are C_1 and C_2, as Figure 14 presents.

Let C_1 and C_2 be continuous differentiable functions, and due to the same width of the lines to be detected, if we select one point P randomly from C_1, and the normal of P intersects with C_2 at P'. Apparently, the P and P' will satisfy the following:

- $W(P) = |P - P'| = W_0$
- An edge has a normal always pointing to their corresponding grey regions which possess lower grey values, therefore, for an edge point P of a baseline represented by a thin grey region, along its normal direction there is a corresponding point P' on the other side of the edge such that the difference between their normal directions is approximately 180° as Figure 14 shows.

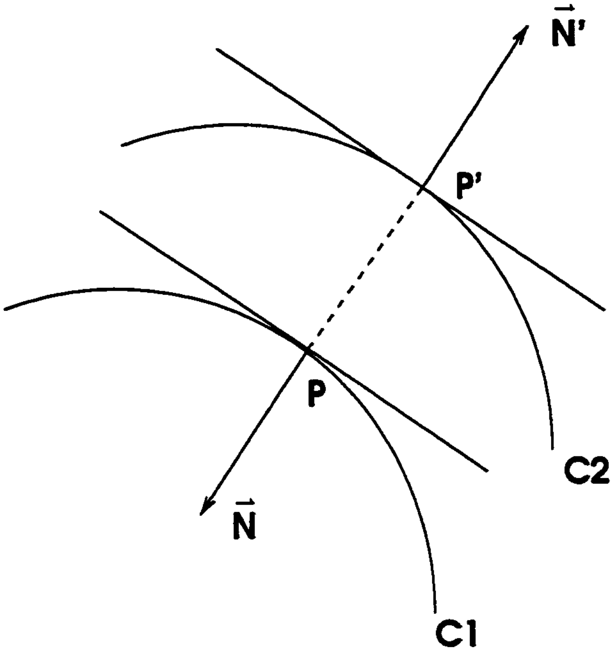


Figure 14: Two smooth arcs and their normal

In a nutshell, the prior rules are the basic constraints that we are following in order to locate the baseline accurately.

Implementation of such an algorithm involves three steps:

- Search for edge point candidates
- Normal calculation of these points
- Accurate location of the edge points

1. Search for edge point candidates:

   Just as mentioned before, we choose Sobel operator to find the candidate edge points in those N thin vertical regions. In addition, we use a 3×3 mask illustrated as Figure 15:

   $\Delta x = (a_{13} + 2a_{23} + a_{33}) - (a_{11} + 2a_{21} + a_{31})$
   $\Delta y = (a_{31} + 2a_{32} + a_{33}) - (a_{11} + 2a_{12} + a_{13})$
   $A = |\Delta x| + |\Delta y|$

a_{11}	a_{12}	a_{13}
a_{21}	a_{22}	a_{23}
a_{31}	a_{32}	a_{33}

Figure 15: 3X3 mask

$\theta = \arctan(\Delta y / \Delta x)$ A is the gradient magnitude of the candidate edge point, θ is the gradient direction. After that, we will eliminate the points with low gradient magnitude by selecting a proper threshold.

2. Inner Normal calculation: Before locating the corresponding edge points along their normal direction, first we must calculate their inner normal. The gradient direction angle(degree) at a pixel calculated by Sobel operator falls within the range of $[-90, 90]$. In order to incorporate the constraint of edge directions in the subsequent processing of the system, this range is extended to $[0, 360]$. Furthermore, the normal directions of local edges have to point at those of their corresponding grey regions with lower grey values. To satisfy the above requirements, the following formula, which is derived by analysing the grey distributions of local edge points [23], is used to convert the gradient direction to inner normal φ:

   - if $\Delta x = 0, \Delta y > 0, \Longrightarrow \varphi = \frac{3}{2}\pi$,

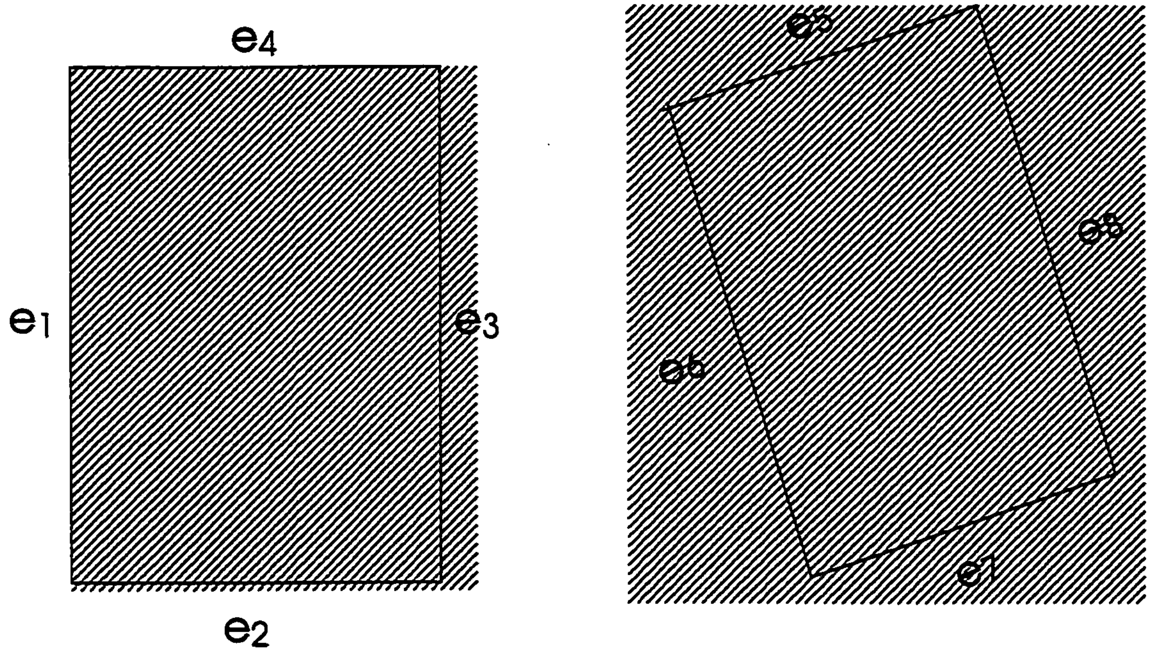


Figure 16: Sample diagram of the region to be processed

if $\Delta x = 0, \Delta y < 0, \Longrightarrow \varphi = \frac{\pi}{2}$.

- if $\Delta y = 0, \Delta x > 0, \Longrightarrow \varphi = \pi$,
  if $\Delta y = 0, \Delta x < 0, \Longrightarrow \varphi = 0$.
- if $\Delta x \cdot \Delta y > 0, and \Delta x > 0 \Longrightarrow \varphi = \pi + \theta$,
  if $\Delta x \cdot \Delta y < 0, and \Delta x < 0 \Longrightarrow \varphi = \theta$.
- if $\Delta x \cdot \Delta y < 0, and \Delta x < 0 \Longrightarrow \varphi = 2\pi - |\theta|$,
  if $\Delta x \cdot \Delta y < 0, and \Delta x > 0 \Longrightarrow \varphi = \pi - |\theta|$.

  If the grey level of this thin region is higher than that of the neighbour region, we just need to rotate the prior φ over 180^o.

3. Location of accurate edge points :

   Now, we can find those edge points expected based on their φ already obtained.

   Suppose P^* is an edge point, φ^* is its inner normal, (x^*, y^*) is the coordination of P^*, ΔW is a reasonable width deviation of the baseline, φ_T is the deviation

Figure 17: Cheque image after applying normal angle constraint

of φ, then we have:

$$x_i = x^* + (W + i)\cos\varphi^*$$
$$y_i = y^* + (W + i)\sin\varphi^*$$
$$-\Delta W < i < \Delta W$$

if such a point P_{i0} exists and satisfies the following:

$$||\varphi_{i0} - \varphi^*| - | - \pi| \leq \varphi_T \tag{23}$$

P_{i0} is assumed to be the corresponding edge point of P^*. In fact, we use the middle point of P_{i0} and P^* as the position of the baseline, and we will still compare the grey levels of these detected baselines to make sure they are the right lines we are looking for as Figure 17 illustrates.

3.6 Complete baselines location

So far, we know the y position of the baselines of legal amount and courtesy amount, then it is somewhat easy for us to obtain their x positions, in other words, the starting

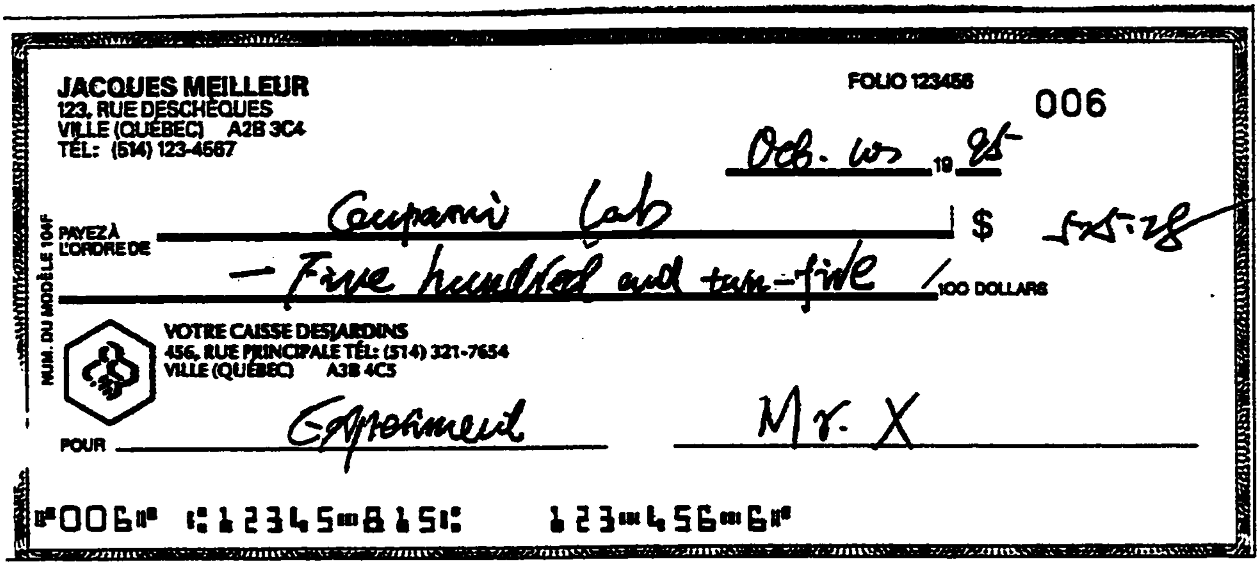


Figure 18: Cheque images with accurate baseline location

point and the ending point of these baselines based on their grey level and length. After that, we can repeat the previous algorithm on the date region to get the date baseline.

Figure 18 shows the accurate position of each baseline of interest.

Until now, we have found the right positions of the baselines, so the next step is to automatically extract those items from cheques we are interested in, including legal amounts, courtesy amounts and dates.

Chapter 4

Extraction of items from cheques

4.1 Layout analysis

In order to extract legal amount, courtesy amount and dates from cheques automatically, a method is proposed to describe cheques based on baselines.

From the previous chapter we can see that a standard Canadian cheque could be described by the legal amount baseline, payee baseline, date baseline, and courtesy baseline. There are other baselines on a cheque, such as those for signature and for noting the purpose of writing the cheque. For some cheques, there are also the baselines which are used for providing the addresses of the owners of the cheques. Therefore, in order to describe a cheque, the legal amount baseline, payee baseline, date baselines, and (or) courtesy baseline must be identified from all the baselines extracted from this cheque. In Chapter 3, we proposed a new method for a fast location of the baselines. Figure 19 shows the final structural description derived from the cheque shown in Figure 1.

After the identification of baselines, a cheque is described by 4-tuples $\{\theta_0, b^{(l)}, x_L^{(l)}, x_R^{(l)}\}$, $\{\theta_0, b^{(p)}, x_L^{(p)}, x_R^{(p)}\}$, $\{\theta_0, b^{(d_1)}, x_L^{(d_1)}, x_R^{(d_1)}\}$ and $\{\theta_0, b^{(d_2)}, x_L^{(d_2)}, x_R^{(d_2)}\}$, and $\{\theta_0, b^{(c)}, x_L^{(c)}, x_R^{(c)}\}$, which correspond to the legal amount baseline, payee baseline, two date baselines, and (or) courtesy baseline, respectively.

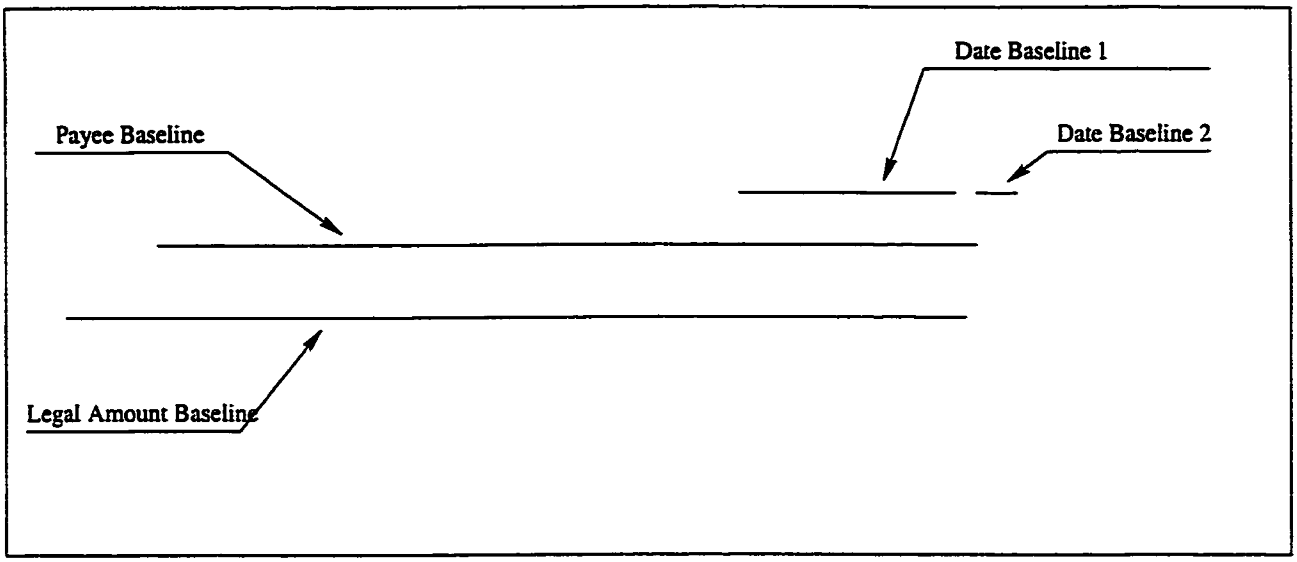


Figure 19: Structural description of a Canadian bank cheque

4.2 Extraction of items from cheques

A general method, called **layout-driven item extraction method**, is proposed for the extraction of items from grey scale cheques. The basic idea is as follows: first a searching region and bounding regions are determined for each item of the legal amount, courtesy amount and date based on the layout description, then the grey scale distributions of the handwritten strokes related to each item are estimated, and the items are extracted by tracing the connected components based on these regions and grey scale distributions. The main components of the method are described in the following sections.

4.2.1 Determination of searching and bounding regions

In order to extract the items of legal amount, courtesy amount and date from a cheque, the searching region and bounding regions are first determinated. The physical meanings of the concepts of searching region and bounding region are: for each item, all its strokes should have their parts inside the corresponding searching region, and the probability of the parts of these strokes going outside the bounding region of this item should be very low. In the succeeding stage of tracing the connected components of handwritten strokes, the initial points for tracing must be chosen from the searching regions and all the points of connected components must be restricted within the bounding regions.

To determine the searching and bounding regions for each item, the knowledge about the way people fill it on a cheque is incorporated. The following takes the legal amount item as an example to show the method of determining the searching and bounding regions. In real-life cheques, the strokes of a legal amount frequently touch or cross the baselines. Based on our research, it can be concluded that for the legal amount, touching or crossing mainly occurs between its strokes and the legal amount baseline. The strokes may also touch the payee baseline, but the probability of the strokes touching and crossing the payee baseline or a baseline other than the legal amount baseline is very low. Based on this reasoning, the searching region $R_b^{(l)}$ for the extraction of legal amounts is defined by

$$R_s^{(l)} = \{(x,y)|(x,y) \text{ satisfies inequalities (27) and (28) }\} \tag{24}$$

$$\tan(\theta_0) * x + b^{(p)} < y < \tan(\theta_0) * x + b^{(l)} \tag{25}$$

$$x_L^{(l)} < x < x_R^{(l)} \tag{26}$$

and

$$R_s^{(l)} = \{(x,y)|(x,y) \text{ satisfies inequalities (30) and (31)}\} \tag{27}$$

$$\tan(\theta_0) * x + b^{(p)} \leq y < \tan(\theta_0) * x + b^{(l)} + D_{pl}/(2 * \cos(\theta_0)) \tag{28}$$

$$0 < x < x_R^{(l)} + T_{bl} \tag{29}$$

where $y = \tan(\theta_0) * x + b^p$ which are the straight line equations of the payee and legal amount baselines, respectively, D_{pl} is the distance between these two baselines, $x_L^{(l)}$ and $x_R^{(l)}$ are the x-coordinates of the left and right end points of the legal amount baseline, respectively, and T_{bl} is a given value. T_{bl} is used to determine the right

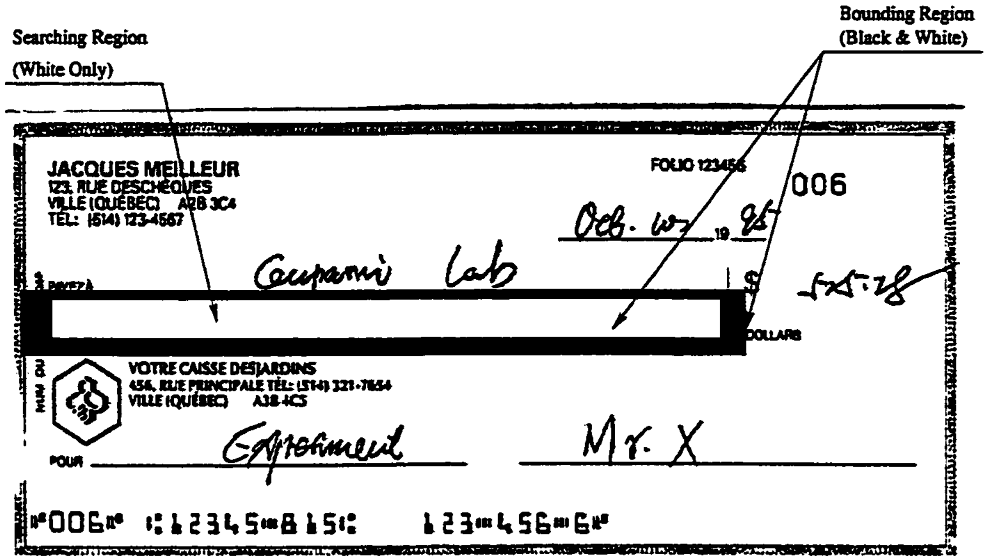


Figure 20: Searching and bounding regions for the legal amount.

boundary for the extraction of the strokes of handwritten cents on cheques. In general, $T_{bl} \geq D_{pl}/2$ guarantees the satisfactory performance of extracting the strokes of legal amount. Figure 20 shows the determined searching and bounding regions for the legal amount on a cheque.

4.2.2 Estimation of the grey distributions of items

The items such as the legal and courtesy amounts and date in a cheque image can be considered as thin connected regions with their grey values lower than those of their local background regions. As mentioned in Chapter 3, an edge has a normal always pointing to their corresponding grey regions which possess lower grey values. Therefore, for an edge point P of a stroke represented by a thin grey region, along its normal direction there is a corresponding point P^* on the other side of the edge such that the difference between their normal directions is approximately 180°. Hence, the regions of the strokes of the legal and courtesy amounts and date can be detected by searching for the corresponding edge points which satisfy the above constraint. Once the points which belong to the thin regions have been detected, the grey distributions of legal amount, courtesy amount and date can be estimated based on the pixel values of these points. The following summarize the steps used to determine the grey scale distribution for an item:

1. Determine a region R which contains the strokes (maybe partial or all) of the item to be extracted. In our system, this region R and the searching region are the same.

2. Detect the points of the item connected regions with their grey values lower than those of their local background regions within R based on the technique of searching for the corresponding edge points. In our system, all the points for searching their corresponding points must belong to the components of the connect edge points whose lengths are longer than a given threshold. Besides, only the medial point of a point and its corresponding point is taken for the calculation of grey scale distribution of handwritten strokes.

3. Calculation histogram $Hist[i], i = 0, ..., 255$, based on the grey scale values of the points within the connected thin regions obtained in the above second step.

Hist obtained in step (3) can be considered as the estimation of the grey distributions of the item, from which, a threshold T for tracing the connected components of handwritten strokes can be determined based on the following formula:

$$\frac{\sum_{i=0}^{T} Hist[i]}{\sum_{i=0}^{255} Hist[i]} \leq \beta \tag{30}$$

where β is a given value within $[0, 1]$. In theory, β can be taken as 1 if all the points detected in the above step 2 belong to the handwritten strokes. In practice, it is recommended that β be assigned from 0.75 to 0.95. In our system, β is assigned a value of 0.90, and the parameter determined for the legal amount item is 84.

4.2.3 Tracing the connected components

Extraction of the item is done by finding the connected components [24] of which the grey values of their connected points are lower than T determined above within

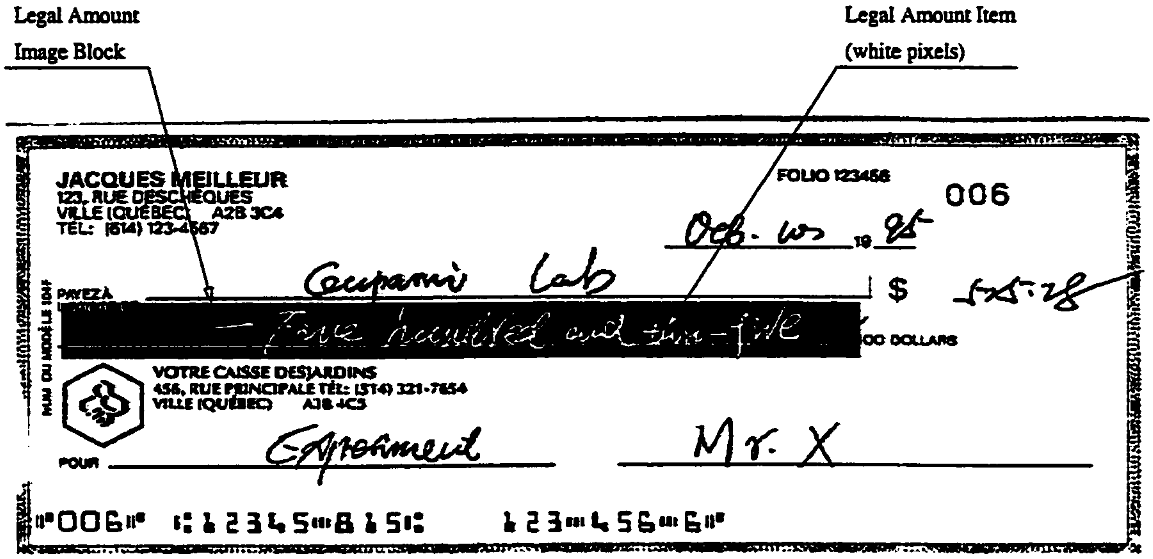


Figure 21: block image of the legal amount

a restricted region. All the initial points for tracing out the connected points must be chosen from the searching region and all the connected points must be restricted within the bounding region. A constraint [16, 25] based on the concepts of *row region straight line segment and column region straight line segment* can be incorporated in the above connected component tracing algorithm to enhance the quality of block images of items. Figure 21 shows the determined block image of the legal amount.

4.2.4 Discussions on the extraction of courtesy amounts

It is clear that the key step in using the above **layout-driven item extraction method** is to determine the searching and bounding regions for the item.

The method of extracting the date is similar to that of the legal amount except that the searching region $R_s^{(d)}$ and the bounding region $R_b^{(d)}$ here are determined by date baselines [16, 25].

The method of extracting the courtesy amount is also determined from the initial searching region $R_s^{(c)}$ and the bounding region $R_b^{(c)}$. However, as can be seen from the following, it will be much more complicated to extract the items of courtesy amount.

Although there is a bounding box for filling the courtesy amount on most cheques, the bounding edge lines cannot be extracted easily due to the low grey contrast against the background. The courtesy amount is sometimes positioned on the right hand side of the legal amount baseline. It may also be located on the right of the payee baseline. However, no matter where it is positioned, there is always a machine printed dollar sign '$' in front of the courtesy amount. Therefore, a courtesy amount item extractor **NumExtr-D** is developed by finding '$' in a cheque image. **NumExtr-D** first locates '$' based on the layout structure of the cheque and the structural description of '$', then determine the searching and bounding regions for the courtesy amount based on the position of '$' and the distance between the payee and legal amount baselines.

In real applications, the strokes of courtesy amount may even touch or cross '$' occasionally. In such cases, **NumExtr-D** cannot be used directly. Based on our experience gained from processing real cheques, we have established the following rules which can be applied to a standard Canadian cheque:

1. 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 courtesy amount lies above this short baseline;

2. If the right end points of both amount and payee baselines almost reach the right edge of the cheque, the courtesy amount lies above the right part of the payee baseline;

3. If the conditions corresponding to the above two rules are not satisfied, the courtesy amount may lie on the right hand side of either the legal amount or payee baseline.

We have developed other four procedures **NumExtr-1**, **NumExtr-2**, **NumExtr-3**, and **NumExtr-4** for the extraction of the courtesy amount, which are used in different situations. **NumExtr-1** is applied if there is a short baseline on the right hand side of the payee baseline and it is also based on the principle of determining the initial searching and bounding regions. **NumExtr-2** is used when the condition corresponding to rule (2) applies. **NumExtr-3** and**NumExtr-4** correspond to rule (3)

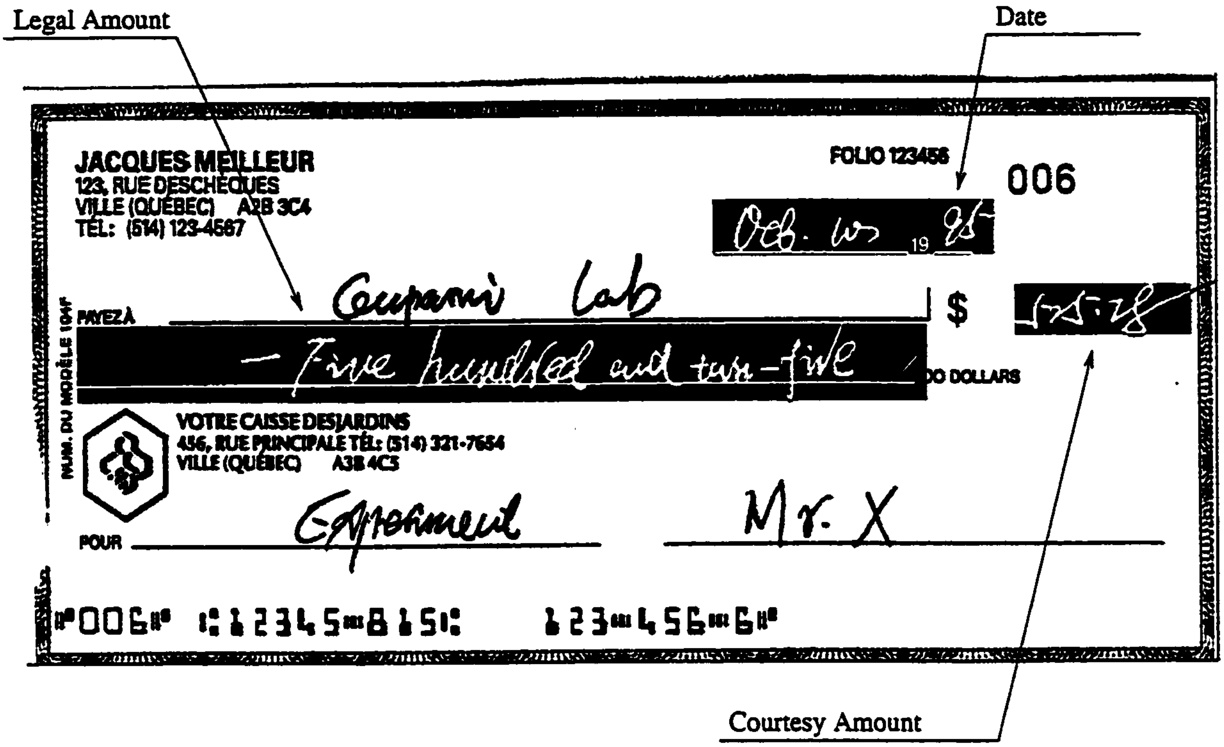


Figure 22: Result of item extraction

above and are used to extract the courtesy amount located on the right hand side of payee baseline and the legal amount baseline, respectively. **NumExtr-2**, **NumExtr-3** and **NumExtr-4** are all developed based on the same principle. The idea is as follows: first extract the connected components of pixel points which have grey values lower than T_c within a specified region, then determine the bounding rectangle of each extracted connected component, and the courtesy amount is finally extracted by grouping the bounding rectangles from the right to the left of the specified region according to the distance between two consecutive bounding rectangles. The algorithms for **NumExtr-3** and **NumExtr-4** are similar. However, in the algorithm for **NumExtr-2**, the pixels on the payee baseline should be excluded to determine the bounding rectangles of the connected components.

A general method of extracting the courtesy amount can be obtained by combining **NumExtr-D**,**NumExtr-1**,**NumExtr-2**, **NumExtr-3** and **NumExtr-4**. Figure 22 shows the legal and courtesy amounts and date extracted from the original image.

4.3 Separation of strokes from connected lines

The connected components extracted from a cheque image may contain both strokes and lines which either touch or cross each other, or the baselines, or occasionally the bounding box lines of the courtesy amount(these cases can be verified by checking if there are baselines inside the block regions corresponding to courtesy amount items). Therefore, post-processing is needed to remove such lines which do not belong to the strokes of the legal and courtesy amounts, and date, from the connected components. An effective technique of separating strokes from connected baselines has been developed based on the morphological closing operation and topological processing [11]. The basic idea to separate a baseline from an extracted item is that the whole baselines are first removed from the item image and then the lost information corresponding to the strokes of data is restored by using both morphological and topological processing.

Suppose $f_{item}(i,j)$ is an item block image and is described by

$$f_{item}(i,j) = \begin{cases} 1 & \text{if (i,j) is point of item} \\ 0 & \text{otherwise} \end{cases} \tag{31}$$

Let $f_{grey}(i,j)$ be the image which corresponds to the sub-image region of $f_{item}(i,j)$ in the original grey cheque image(see fig. 8). Assume that **B1** is a baseline inside the sub-image region that corresponds to the item block image in the original grey cheque image. Because we know the position and width of **B1**, it is easy to determine a sub-image region bl_{region} in $f_{item}(i,j)$ or $f_{grey}(i,j)$ where all points of **B1** lie. Then our method of separating the strokes of item from **B1** is summarised in the following steps:

1. Elimination of baseline

2. Morphological closing operation

3. Topological processing

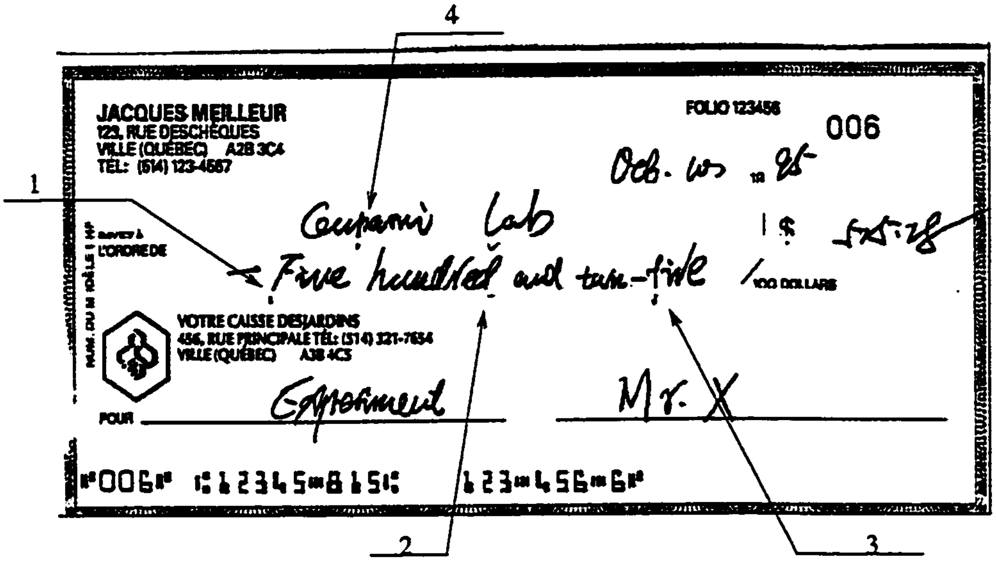


Figure 23: lost information after baselines are removed

4.3.1 Elimination of baseline

Given that the background has been completely eliminated from the gray-scale image and baselines are located, elimination of baselines from the gray-scale image is performed as follows:

$$f_{item}^{(e)}(i,j) = \begin{cases} 0 & \text{if(i,j)} \in Bl_{region} \\ f_{item}(i,j) & \text{otherwise} \end{cases} \tag{32}$$

Unfortunately, as Figure 23 illustrates, when the baselines are cut from the gray-scale image there is a loss in the handwritten information that intersected with the baselines before. As a remedy, we will introduce a formal approach using morphological processing followed by topological processing to restore (fill the gaps) the lost information.

4.3.2 Why Morphology?

The purpose of using morphological processing here [26] is to increase the difference in the intensity between a baseline and the handwritten data that intersects with that baseline.

As we pointed out in the previous sections, we aim to extract handwritten data from gray-scale bank cheques without damaging data after the baselines are removed. Based on our observation, when the intensity of handwritten data is close to the intensity of the baselines, the situation becomes more difficult. Moreover, if the handwritten data and the baselines intersect, then the elimination of baselines and the preservation of the handwritten data becomes a non-trivial problem. In fact, our purpose is to use the morphological closing operation to increase the difference in the intensity between a baseline and the handwritten data that intersects with that baseline. Technically, this means that we are propagating the intensity of the handwritten data as well as the intensity of the background over the intensity of the baselines. As a result of this propagation, it is easy to threshold the baseline from the morphological image (cheque in our case) preserving the handwritten information that intersects with the thresholded baseline. Figure 24 illustrates the results of performing a closing operation on a cheque image. Without loss of generality, we should mention the most important role played by the size and shape of the structuring element in increasing the contrast between the baseline and the handwritten data. From an implementation point of view, 3x3 structuring element(011,011,000) with 2 iterations is adopted in this context.

4.4 Morphological Processing

Given f_{filt}, we will use morphological processing[said] to restore the lost handwritten information that intersects with the eliminated baselines. To restore this lost information we will perform a morphological closing operation as in [27, 28] on f_{filt}, threshold the baselines from the morphological image, and extract the information within the line's region to be added to f_{lr}. In this section, we will present the morphological closing operation and its important property as in [27] before we deal with the other processes mentioned in this paragraph.

The gray-scale closing is defined as

$$f \bullet V = (f \oplus V) \ominus V \tag{33}$$

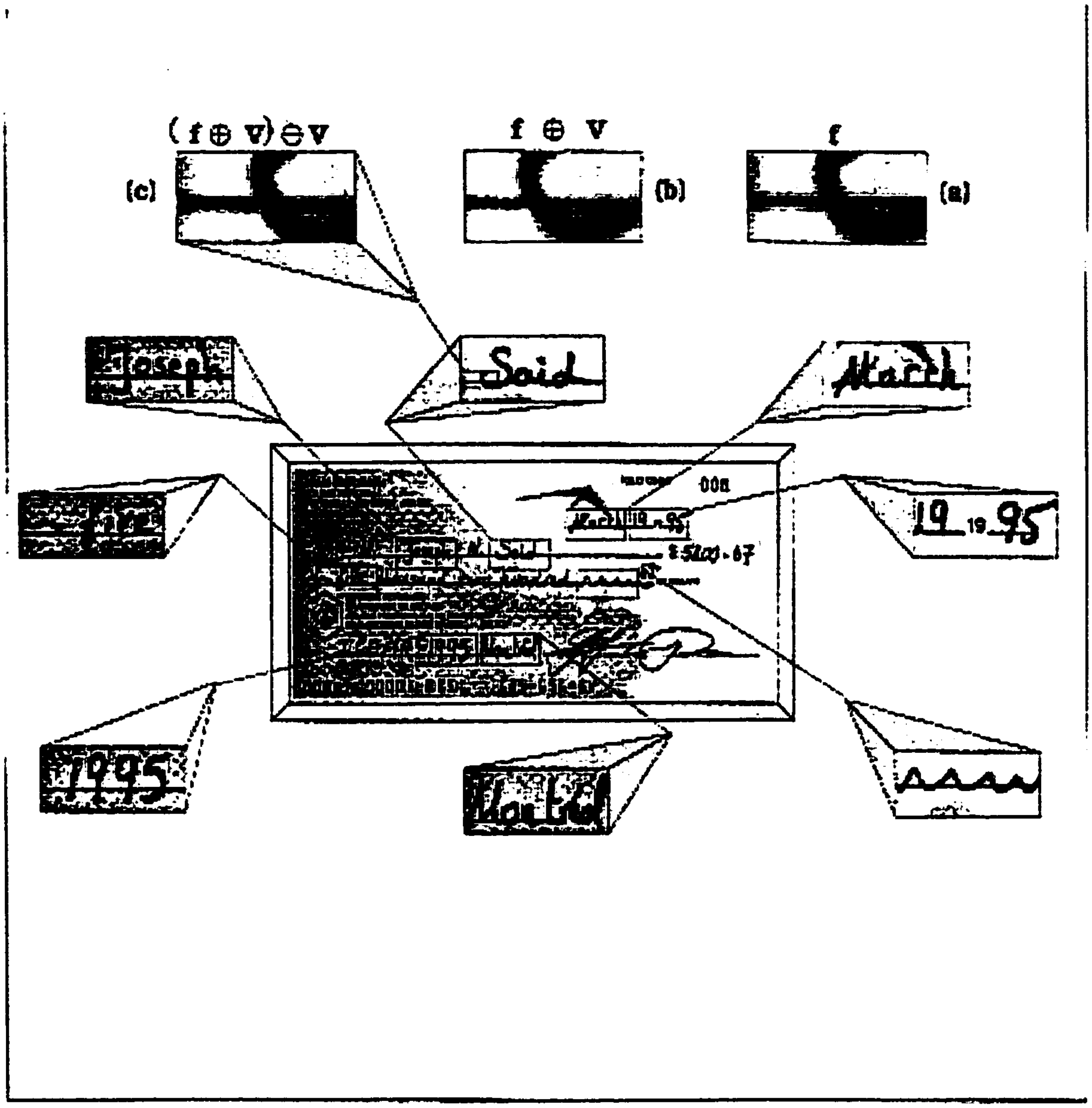


Figure 24: Morphological closing operation performed on image

where

$$(f \ominus V)(i,j) = min\{f(i+k, j+l) | (k,l) \in V_{\triangle}\} \tag{34}$$

$$(f \oplus V)(i,j) = max\{f(i+k, j+l) | (k,l) \in V_{\triangle}\} \tag{35}$$

are the erosion and the dilation respectively of image f by the structuring element V.

Closing has an interesting property: it is an *idempotent* transform; once it has been applied, it is useless to apply it again. This is defined as:

$$f \bullet V \bullet V = f \bullet V. \tag{36}$$

According to [27], every nonlinear transformation results in some loss of information; but if a transformation is idempotent, the amount of information it would lose is self-controlled which is an important property to be considered in our case. The resulting image is named f_{clos}

4.5 Restoring Lost Information

Knowing the positions and lengths of the baselines, we can now process the morphological image f_{clos}. This is achieved by thresholding baseline pixel values according to their mean M using Equation 39 , and combining the results with the image f_{lr} as follows:

$$f_{clos}(i,j) = \begin{cases} f_{clos}(i,j) & \text{if } f_{clos}(i,j) \leq M \\ 255 & \text{otherwise} \end{cases} \tag{37}$$

$$f_{union}(i,j) = \begin{cases} f_{clos}(i,j) & \text{if } (i,j) \in L_{Sk} \\ f_{lr} & \text{otherwise} \end{cases} \tag{38}$$

Figure 25 show the information gained after a morphological closing operation is performed.

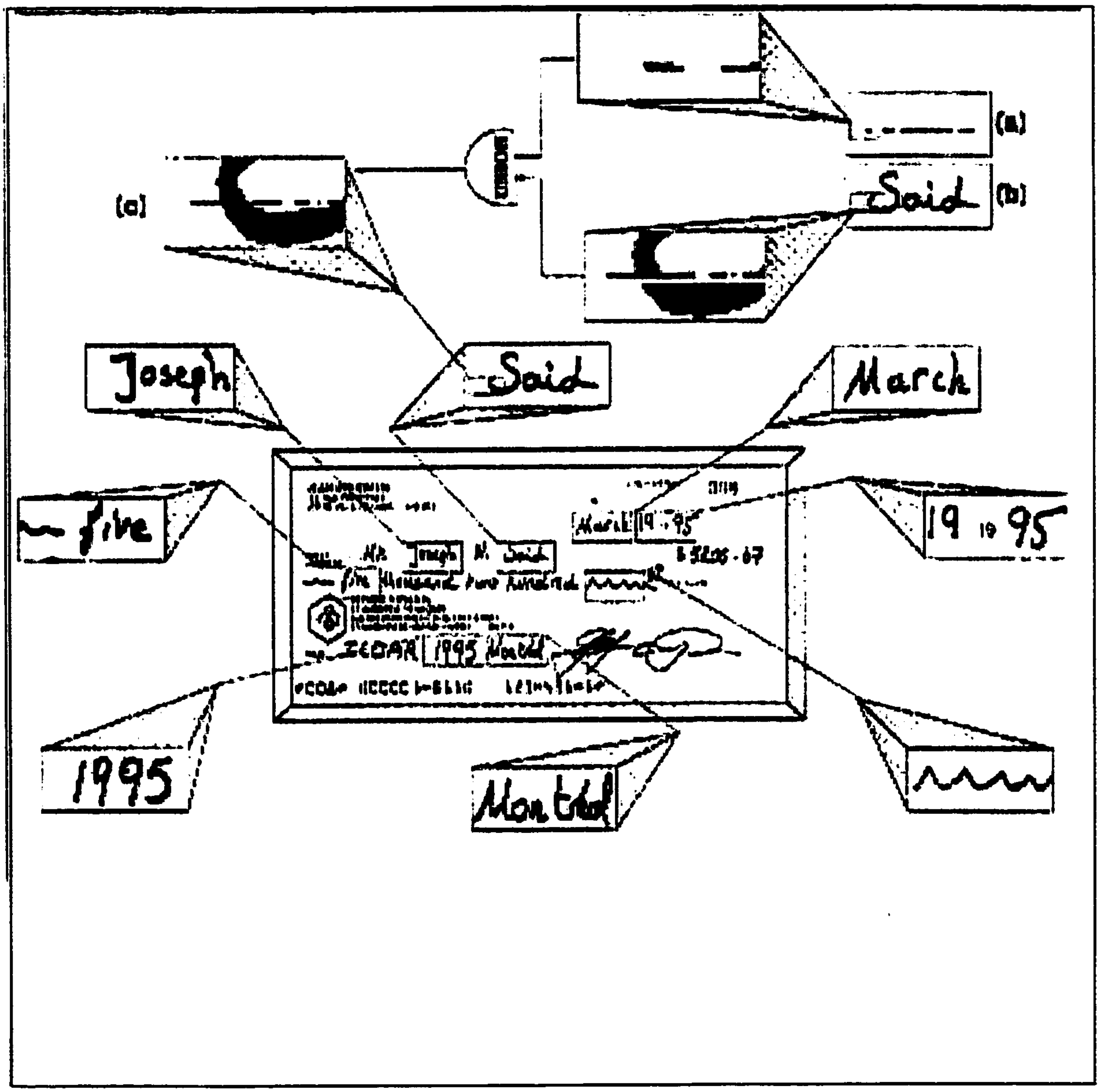


Figure 25: Restoring the lost information that intersects with the eliminated baselines

4.5.1 Mathematical Morphology

After introducing mathematical morphology and showing that the morphological closing operation is needed in our implemention and how it helped in reducing the amount of lost handwritten information that intersects with the baselines on the cheques, it is important to ask the following question: Is it enough to use the closing operation to capture all the lost information and terminate the process by extracting the handwritten information? Unfortunately, the case is not as easy as one thinks. Further processing should be performed because there are cases where mathematical morphology does not restore all the lost information. In the coming sections, we will introduce *topological processing* as a solution to fill the gaps which were not restored by mathematical morphology in order to restore as much information as possible after thresholding the baselines in the morphological image. But before we proceed we need to apply median filtering for noise elimination and image recovery on f_{union} as follows:

$$f_m(i,j) = Median\{f_{union}(i-k, j-l)\forall k,l \in W\}, \tag{39}$$

where W is a suitably chosen window. In our work, a 3 x 3 window is employed.

4.6 Topological Processing

As mentioned in section 4.5, the purpose of morphological processing, namely the closing operation, is to reduce the intensity of baselines and increase the intensity of the handwritten data in order to minimize the loss of useful data resulting from a threshold that eliminates the baselines in the gray-scale morphological image. Unfortunately, this approach does not always completely restore the handwritten data that intersects with the baselines. In such case, thresholding the baselines from f_{lr} will result in a loss in the topological properties of the handwritten information. As a solution to this problem, more intelligent processing should be applied to recover as much as possible the unfilled gaps. In pursuing this line, our algorithms identified the existence of the unfilled gaps between their corresponding extremities in the images and used an automated methodology to establish connections between these extremities to restore the lost information by filling the gaps. In fact, the closing operation contributes to enhancing the gaps in the line's region and makes the process of gap identification and filling easier.

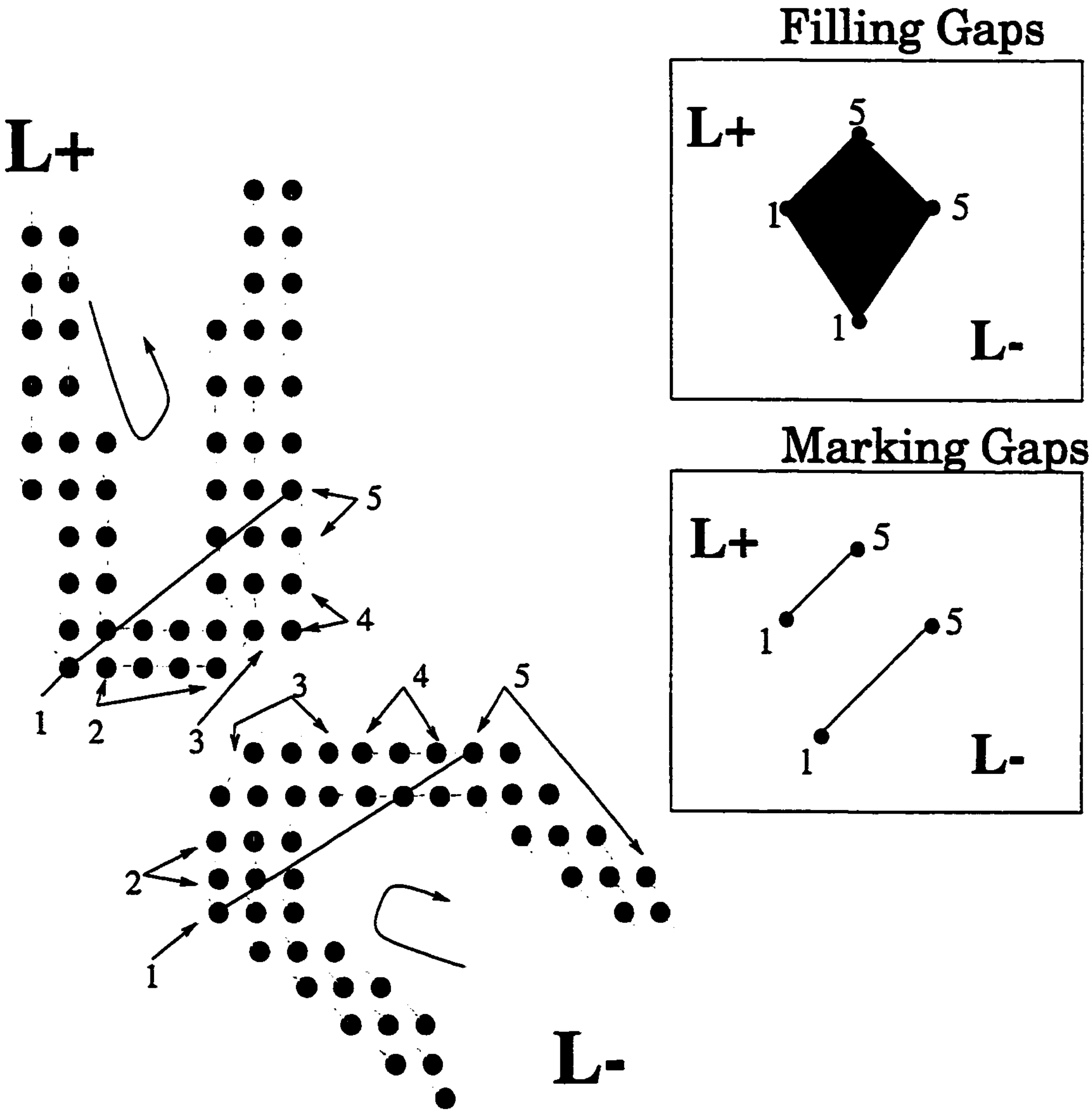
4.6.1 Identification and Filling of Gaps

Three possible configurations of gaps have been considered according to the information lost with diagonal handwriting or perfect superposition. The four extremities that delimit gaps are labelled: L+, L-, R+, and R- as illustrated in Figure 26, 27, 28. Identification of gaps is achieved by first performing an edge detection algorithm on the image f_{union} with the purpose of finding the edges of the handwritten data which were lost due to thresholding the baselines in the morphological image f_{lr} and not restored in f_{clos} using the morphological closing operation. After edge detection, the gradient direction angle α at a pixel is replaced by θ calculated by the following formula:

$$f_{union}(i,j) = \begin{cases} 90 & \text{if } dx = 0 \text{ and } dy < 0 \\ 270 & \text{if } dx = 0 \text{ and } dy > 0 \\ 180 & \text{if } dy = 0 \text{ and } dx > 0 \\ 0 & \text{if } dy = 0 \text{ and } dx < 0 \\ 180 + \alpha & \text{if } dx > 0 \text{ and } dy < 0 \\ 360 + \alpha & \text{if } dx < 0 \text{ and } dy > 0 \\ 180 + \alpha & \text{if } dx > 0 \text{ and } dy < 0 \\ \alpha & \text{if } dx < 0 \text{ and } dy < 0 \end{cases}$$

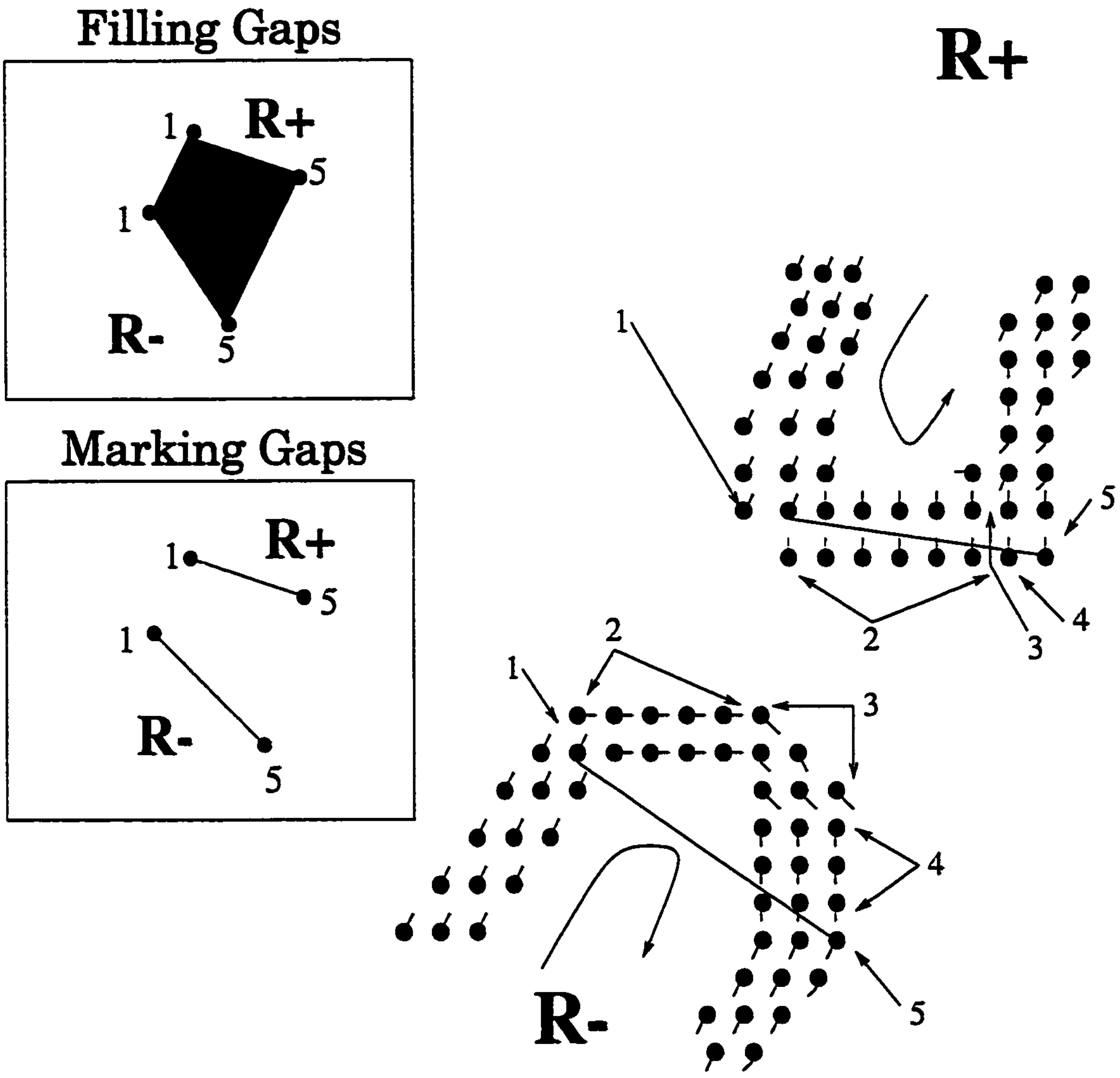
During the edge detection process, we will gather information about the magnitude and the digital gradient of each pixel; this will greatly help us to identify the existence of all unfilled gaps between the corresponding extremities.

Figure 26, 27, 28, 29 demonstrates the extremities $L+$, $L-$, $R+$, and $R-$ and the rules in finding them when they exist based on the calculated normal directions of pixels. Consider, for example, the rule for identifying an L- (Figure 29). There are five stages to pass through in order for the algorithm to declare that in order for the



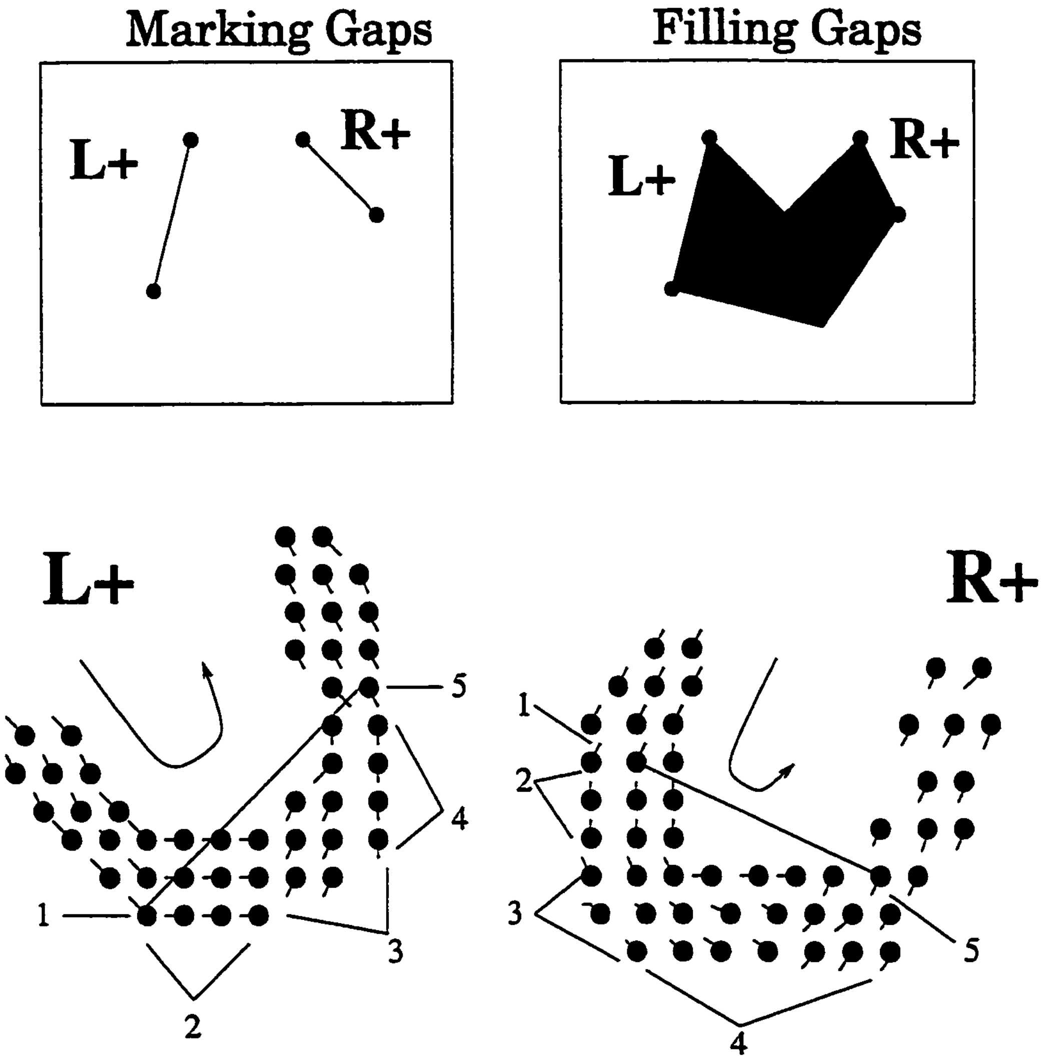
Rule 1

Figure 26: Identifying Gaps between L- and L+



Rule 2

Figure 27: Identifying Gaps between R- and R+



Rule 3

Figure 28: Identifying Gaps between L+ and R+

Identifying an L-

```
          READ EDGE IMAGE ;
          COUNT = 0;
LOOP: READ θ OF ONE PIXEL;
      IF (180<θ <270);
         IF (135<θ <225)
            IF (90<θ <180)
               IF (45<θ <135)
                  IF (0<θ <90)
                    ONE L- FOUND;
                       GOTO NEXT PIXEL;

                  ELSE COUNT++;
                   IF(COUNT>M)?:EXIT:NEXT PIXEL;
               ELSE COUNT++;
                   IF(COUNT>M)?:EXIT:NEXT PIXEL;
            ELSE COUNT++;
                IF(COUNT>M)?:EXIT:NEXT PIXEL;
         ELSE COUNT++;
             IF(COUNT>M)?:EXIT:NEXT PIXEL;
      ELSE COUNT++;
          IF(COUNT>M)?:EXIT:NEXT PIXEL;
EXIT

NOTE: M IS A THRESHOLD, AND θ IS THE NORMAL DIRECTION OF PIXEL
```

Figure 29: Identification of an L- given the image f_{edge}

algorithm to declare that an L+ is found. In step 1, after a pixel is located with a normal direction $\theta(180° < \theta < 270°)$, the algorithm starts to search for a neighbouring pixel θ such that $225° < \theta < 315°$. When such a θ is read, the algorithm knows that step 2 is reached. Similarly, when steps 3, 4, and 5 are identified by the algorithm, the extremity L+ is found where a line is drawn between the pixel discovered in step 1 and the pixel discovered in step 5. The other extremities, L-, R+, and R-, are found following a similar concept. Having identified all possible extremities, gaps are filled according to the following rules in the order: (L+, L-), (R+, R-), and (L+, R+) . The image produced by this process is called f_{gaps}.

It should be mentioned that the rules for identifying L+, L-, R+, and R- are based on the assumption that the baselines are horizontal. In real implementation,

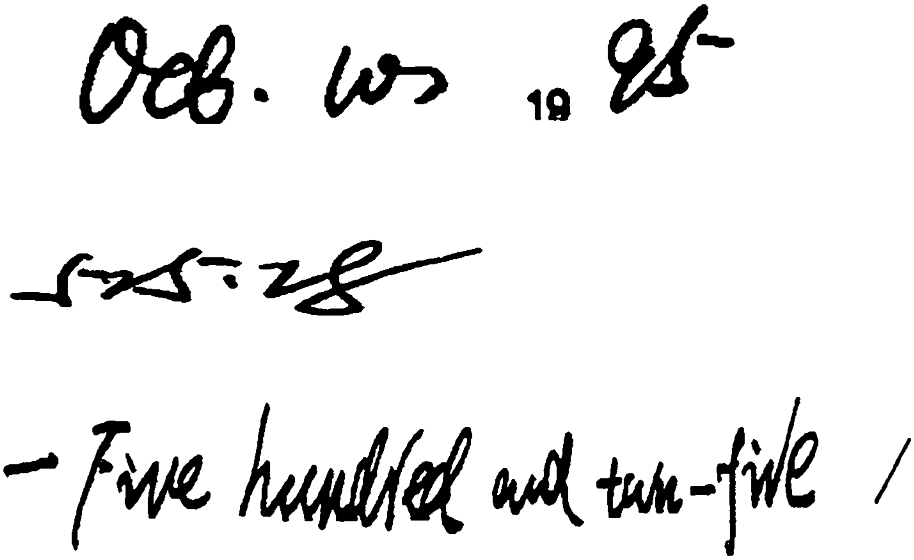


Figure 30: Result of item extraction

the above rules will be applied to the deviation angles between the calculated normal directions of edge points and those of the baselines direction.

Figure 30 shows the item block item images of the legal amount, courtesy amount and date on the cheque obtained finally shown in Figure 1 after line removal, which will be used as the inputs of the data recognition models.

Chapter 5

Experimental Results

5.1 Testing Enviroment and Database

Based on the proposed approach, techniques and algorithms(see Figure 31), an automatic cheque item extraction system has been developed in the Sun-Sparc/SunOS/X-Windows enviroment. The input of the system is a scanned grey level cheque image which has a TIFF format, and the output provides the legal amount, courtesy amount and date items.

Totally we got 1800 real-life cheque images from IRIS database and these cheque images are divided into three sets: one for training(300 check images), one for repeatedly testing our system(300 check images), and one for final testing which has 1200 grey level real-life cheque images.

Actually those 1200 cheque images for the final testing are divided again into two groups **TEST1** and **TEST2**, each of them contains 600 cheque images and our system never "saw" these images before. Tables 1 to 4 provide the results of processing two groups **TEST1** and **TEST2**. The experiments showed that all the baselines on these 600 cheque images were correctly extracted. Tables 1 and 3 illustrate the results of layout analysis for the cheque images of **TEST1** and **TEST2**, respectively. The structural descriptions of 18 cheques in **TEST1** and 13 cheques in **TEST2** were not derived by the system due to broken, missing or distortion of baselines on these cheque images.

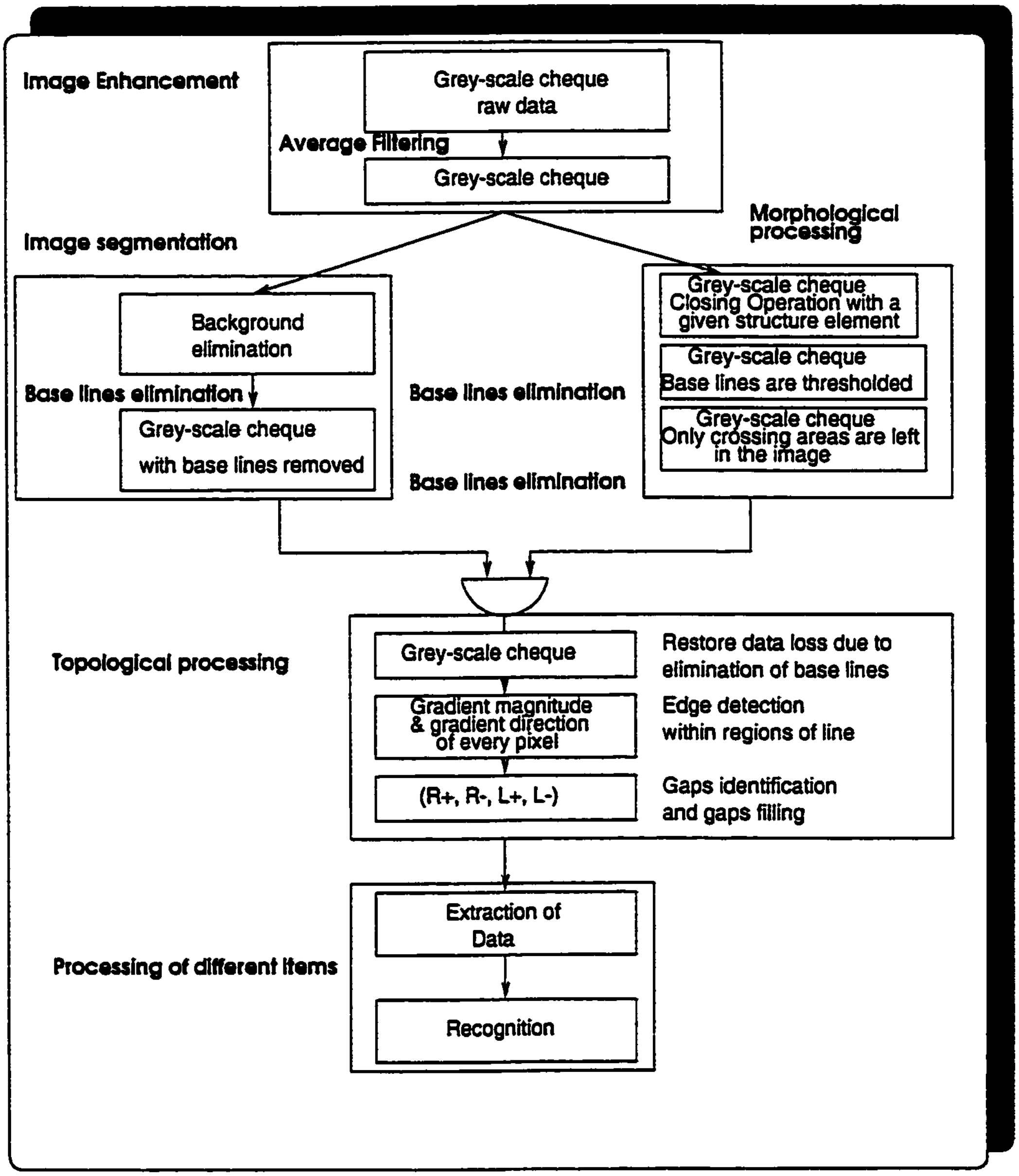


Figure 31: Processing of bank cheques (system design)

Structural description	Correct	Substitution	Rejection	Reliablility
600 cheques	97.00%	0	3.00%	100%

Table 1: Testing results on the structural description of cheques of TEST1.

Extracting items	correct_items	wrong_items
legal amounts	99.46%	0.54%
courtesy amounts	97.85%	2.15%
date	99.23%	0.77%

Table 2: Testing results on 582 (600 X 97.00%) real bank cheque images of TEST1.

In Tables 2 and 4, the results of extracting the items from 582 cheques of **TEST1** and 587 cheque images of **TEST2**, respectively, are provided, where **correct_itmes** and **wrong_items** indicate that the items which cannot be correctly located and extracted. It can be concluded from Tables 1 to 4 that the proposed approach, techniques and algorithms are very effective.

As you may notice that the **wrong_items** rate for courtesy amounts extraction is a little bit high, the reason behind that is the position of the courtesy amount is not fixed, sometimes it is positioned on the right hand side of legal amount baseline and it may also be located on the right side of the payee baseline.

Structural description	Correct	Substitution	Rejection	Reliablility
600 cheques	97.80%	0	2.20%	100%

Table 3: Testing results on the structural description of cheques of TEST2.

Extracting items	correct_items	wrong_items
legal amounts	99.39%	0.61%
courtesy amounts	98.18%	1.82%
date	97.75%	2.25%

Table 4: Testing results on 587 (600 X 97.80%) real bank cheque images of TEST2.

Structural description	Correct	Substitution	Rejection	Reliablility
600 cheques	97.50%	0	2.50%	100%

Table 5: Testing results on the structural description of cheques of TEST1.

5.2 Performance Analysis

5.2.1 Existing system

There is an existing system called **Automated Payment Extraction System(APES)** developed by **CENPARMI** for the extraction of the items of the legal and courtesy amounts and dates from on-line scanned cheque images and we already briefly introduced it in the chapter 2. This system has been tested and demonstrated in many occasions (such as IRIS Annual Conference, June 1996, Montreal, Canada, etc). I test this system **APES** by using the same testing set(1200 cheque images)for my program, from the Table 5-8 we can see that this bank cheque items extraction system works well. But the problem is the processing time of this system is not so satifying, we will address this problem in the next section.

Extracting items	correct_items	wrong_items
legal amounts	99.49%	0.51%
courtesy amounts	97.95%	2.05%
date	99.49%	0.51%

Table 6: Testing results on 585 (600 X 97.50%) real bank cheque images of TEST1.

Structural description	Correct	Substitution	Rejection	Reliablility
600 cheques	98.00%	0	2.00%	100%

Table 7: Testing results on the structural description of cheques of TEST1.

Extracting items	correct_items	wrong_items
legal amounts	99.39%	0.61%
courtesy amounts	98.98%	1.02%
date	97.96%	2.04%

Table 8: Testing results on 588 (600 X 98.00%) real bank cheque images of TEST1.

Methods	Number of cheque images	Total time used (second)
APRS	400	2460
My program	400	1210

Table 9: Testing results on 400 real bank cheque images of TEST1.

Methods	Number of cheque images	Total time used (second)
APRS	400	2560
My program	400	1240

Table 10: Testing results on 400 real bank cheque images of TEST2.

5.2.2 Comparison of two methods

In order to compare the performance between my program and the existing system developed in **CENPARMI**, I designed a batch processing method so that we can get the time they consumed during the procedure of extraction fairly and effectively.

First we carefully select 800 cheque images from IRIS database and divide them into two sets, each one has 400 cheque images. Actually we selected these 800 cheque images based on the following criteria:

- they contain simple and complex background cheque images both;
- they could all be extracted by both systems without reject;
- we don't count whether the extraction result is correct or not
- they contains both formal handwritting and informal handwritting.

We will **start** to calculate the time when these two extraction programs read the cheque image file in TIFF format and **stop** the calculation when those three items legal amount, courtesy amount and date are extracted and written into file.

Tables 9 and 10 indicate the performance between these two extraction programs.

APES stands for **Automated Payment Extraction System** As we can see from Figure 32, in the preprocessing stage, this system first employs the Sobel edge

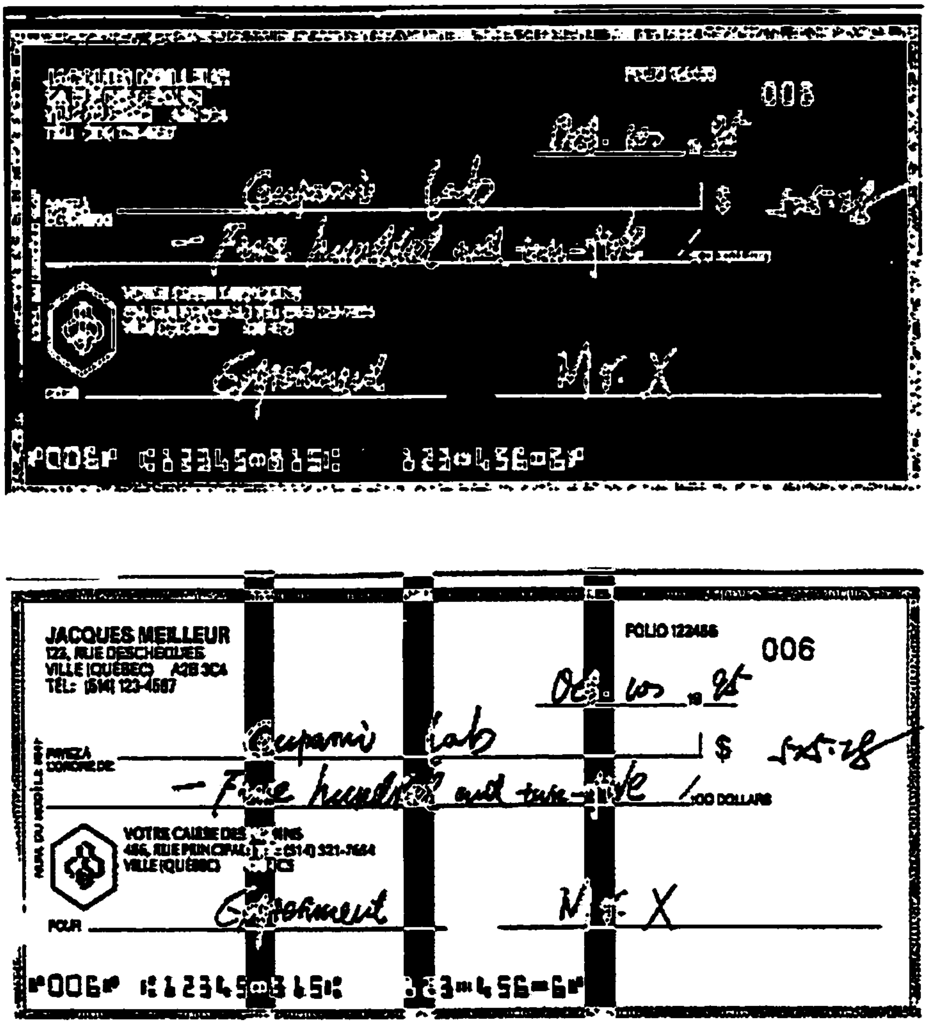


Figure 32: Difference of two algorithms

detector to calculate every point of the cheque image to get its gradient images (both edge strengh and normal direction images), and then the least square fitting and Hough transform are proposed to find and refine the straight lines the from edge images. It is very easy to get to the point that this kind of method would be very time-consuming and will affect the performance of the system apparently.

5.2.3 Some error extraction

In the meantime, we can also find that during the stage of item extraction, some incorrect extraction occurs, they can be classified into four main categories:

- The extraction output may contain noise from the payee part(Figure 33);
- The extraction is incomplete(Figure 34);
- Legal amount strokes may reach the above payee item;(Figure 35)
- The extraction output from the legal amount may contain noise crossed by the strokes from the payee part(Figure 36)

Since these errors could terribly affect the extraction output and especially the later work of segmentation and recognition, so they will be of great importance for the future research work.

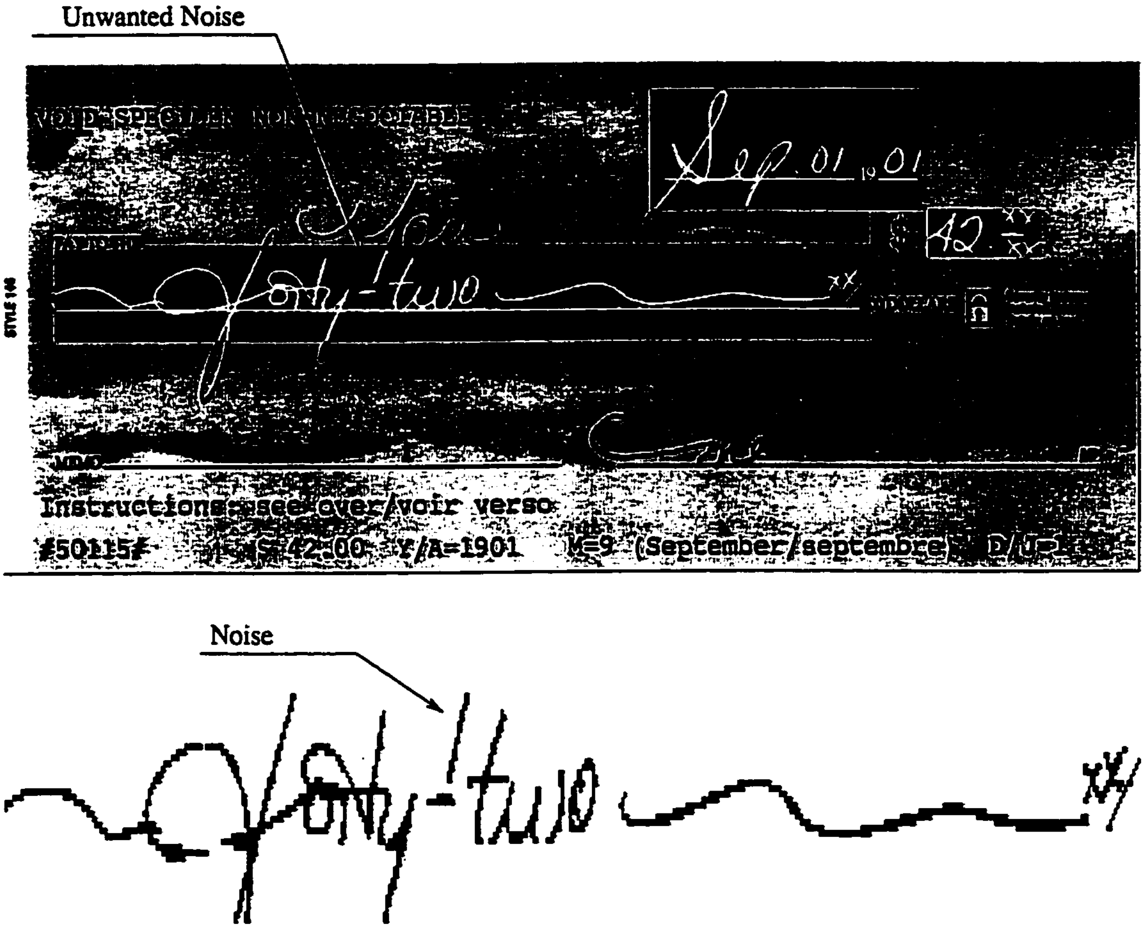


Figure 33: The connected components extracted from the legal amount may contain noise such as part of the name of the payee

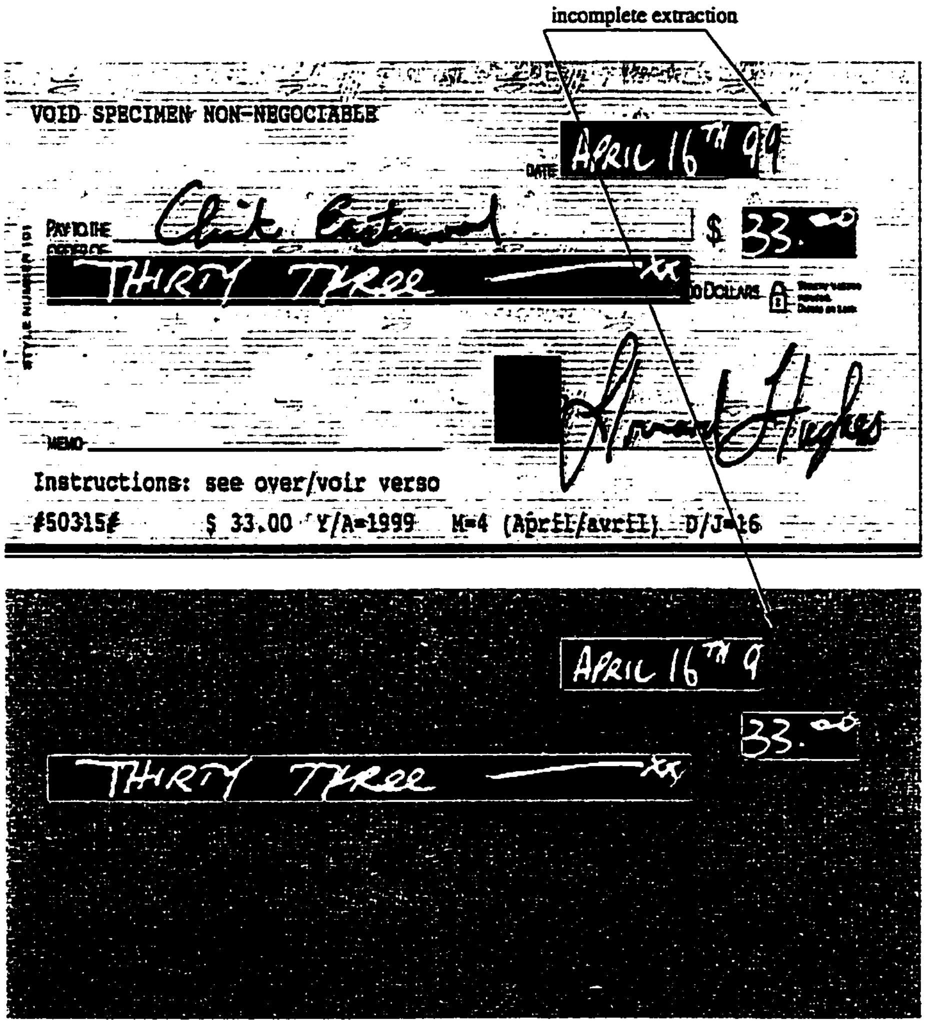


Figure 34: Incomplete extraction

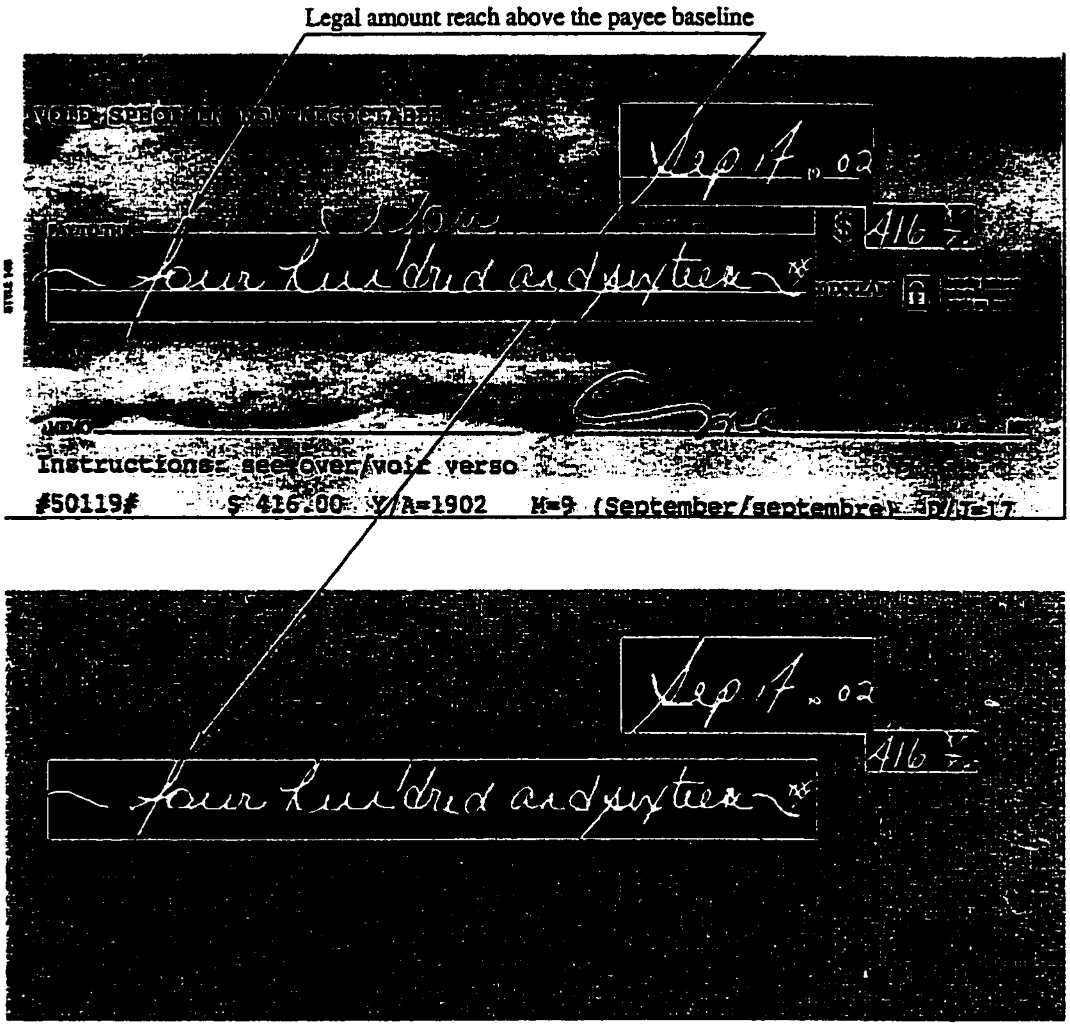


Figure 35: Strokes of a legal amount reach above the payee baseline

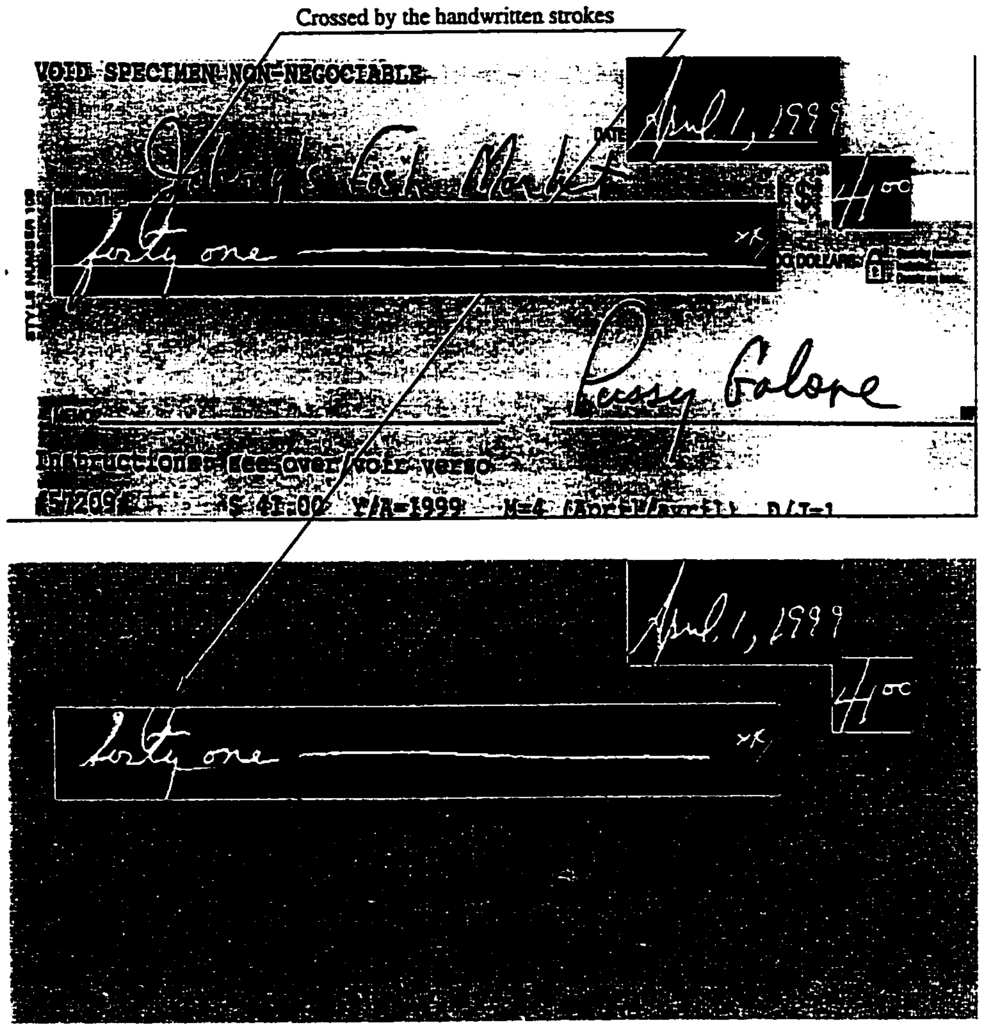


Figure 36: The connected components extracted from the legal amount contain noise crossed by the handwritten strokes

Chapter 6

Conclusion and Future Works

Automatic reading of bank cheques is a very active topic in Document Analysis, one of the most important research issues in automatic reading of cheques is to extract interested items such as legal amount, courtesy amount and date etc. from the cheque images. However, to develop an effective item extraction system is a difficult task, especially when the processed cheques have different styles and contain complicated background pictures.

In this thesis, we have proposed a system that can extract cheque items based on a quick way of determining the baselines of cheques, a-priori information about the positions of the legal and courtesy amounts and date, and a layout-driven item extraction method. Several new image processing techniques and algorithms are proposed, based on which a complete bank cheque item extraction system has been developed in the Sun-Sparc/SunOS/X-Windows envrionment.

The results of a series of experiments indicate that the proposed approaches, techniques and algorithms are effective.

Although our program is mainly proposed for Canadian cheques, it can also be applied or extended to the cheques of other countries. In general, our approach should be applicable to the cheques other than Canadian whose layout structures can be determined by baselines.

6.1 Major Contributions of the Thesis

Automatic banking reading is a challenging field of research. The major contribution of this thesis is to provide a fast way to locate the baselines and then extract items in three regions we are interested from grey level cheque images. It includes three main steps:

- First, recursive dynamic thresholding technique is introduced to eliminate the background from the grey level cheque image;
- Second, baselines that could intersect the handwritten data are recognized by a new fast and effective method according to the normal constraint described in chapter 3.
- After the baselines are removed, morphological and topological processes are proposed to identify and fill the gaps that brought by the elimination of the detected baselines.

Generally, our program has the following merits:

- It has been designed to extract items in various cheques, other than Canadian only.
- It works well not only on cheques with simple backgrounds but also those cheques with complicated backgrounds. The common existing methods normally are only useful in those applications where only specific styles of cheques with simple background pictures are used.
- Morphological and topological techniques highly enhance the accuracy and efficiency of item extraction and improved the quality of the extracted items.

6.2 Future Work

For such a comprehensive set of algorithms, naturally there are many challenges. To realize a high performance cheque item extraction system for commercial use, future study can proceed in the following directions:

- When the background of the cheque has intensities very close to those of the handwritten information, the initial process of background removal would be very difficult.
- The connected components extracted from the legal amount may contain unwanted noise such as part of the name of the payee touched or crossed by the handwritten strokes. When this happens, how do we identify and remove such strokes?
- The morphological and topological operations have to be made more efficient.

Bibliography

[1] L.Lam, C.Y. Suen, et al. Automatic processing of information on cheques. In *IEEE Int. Conf. on Systems, Man, Cybernetics*, pages 2353–2358, Vancouver, Oct. 1995.

[2] Y.Y Tang, C.Y. Suen, and C.D. Yan M. Cheriet. Financial document processing based on staff line and description language. In *IEEE Transactions on Systems, Man, Cybernetics*, pages 738–754, 1995.

[3] T.M. Ha and H. Bunke. Model-based analysis and understanding of cheque forms. In *International Journal of Pattern Recognition and Artificial Intelligence 8, No. 5*, pages 1053–1080, 1994.

[4] D. Guillevic and C.Y. Suen. Cursive script recognition applied to the processing of bank cheques. In *Proc. of the Third Int. Conf. on Document Analysis and Recognition*, pages 11–14, Montreal, Canada, 1995.

[5] J.P. Dodel and R. Shinghal. Symbolic/neural recognition of cursive amounts on bank cheques. In *Proc. of the Third Int. Conf. on Document Analysis and Recognition*, pages 15–18, Montreal, Canada, 1995.

[6] T. Paquet and Y. Lecourtier. Automatic reading of the literal amount of bank checks. In *Machine Vision and Applications 6*, pages 151–162, 1993.

[7] M. Leroux, J. C. Salome, and J. Badard. Recognition of cursive script words in a small lexicon. In *Proc. of the First Int. Conf. on Document Analysis and Recognition*, pages 774–782, Saint-Malo, France, 1991.

[8] E. Lethelier, M. Lerour, and J. F. Couthouis. An automatic reading system for handwritten numeral amounts on french checks. In *Proc. of the Third Int. Conf. on Document Analysis and Recognition*, pages 92–97, Montreal, Canada, 1995.

[9] A. Gupta, W.V. Nagendraprasad, A. Liu, P.S.P. Wang, and S. Ayyadurai. An integrated architecture for recognition of totally unconstrained handwritten numerals. In *International Journal of Pattern Recognition and Artificial intelligence 7, No. 4*, pages 757–773, 1994.

[10] A. Agarwal, L. Granowetter, K. Hussein, A. Gupta, and P.S.P. Wang. Detection of courtesy amount block on bank checks. In *Proc. of the Third Int. Conf. on Document Analysis and Recognition*, pages 748–751, Montreal, Canada, 1995.

[11] M. Cheriet, J.N. Said, and C.Y. Suen. A formal model for document processing of business forms. In *Proc. of the Third Int. Conf. on Document Analysis and Recognition*, pages 210–213, Montreal, Canada, 1995.

[12] C.Y. Suen. The research and development of the automatic payment recognition system. In *Technical Report, CENPARMI*, Concordia University, Montreal, Canada, December 1995.

[13] J. V. Moreau, B. Plessis, O.Bougeois, and J. L. Plagnaud. A postal check reading system. In *Proc. of the First Int. Conf. on Document Analysis and Recognition*, pages 758–766, Saint-Malo, France, 1991.

[14] D. S. Bloomberg. Multiresolution morphological approach to document image analysis. In *Proc. First Int, Conf. on Document Analysis and Recognition*, Sain-Malo, France, Sept. 30-Oct. 2 1991.

[15] Machine Vision and Application. In *Special Issue: Document Image Analysis Techniques*, volume 5, Sept. 30-Oct. 2 1992.

[16] K. Liu, C.Y. Suen, and C. Nadal. The development of cheque image segmentation technique. In *Technical Report, CENPARMI, Concordia University*, Montreal, Canada, 1995.

[17] A.K. Jain. *Fundamentals of Digital Image Processing.* Prentice Hall, Inc., Englewood Cliffs, New Jersey 07632, USA, 1989.

[18] J. Illingworth and J. kittler. A survey of hough transform. In *Computer Vision, Graphics, and Image Processing*, volume 44, pages 87–116, 1988.

[19] D.C.W. Pao, H.F. Li, and R. Jayakumar. shape recognition using the straight line hough transform: theory and generalization. In *IEEE Trans. Pattern Analysis and Machine Intelligence*, volume 14, pages 1076–1089, 1992.

[20] N. Otsu. A threshold selection method from gray-level histograms. In *IEEE Trans. Systems Man Cybernet.*, volume SMC-8, pages 62–66, 1978.

[21] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* New York:Academic, 1972.

[22] Svetha Venkatesh. Edge evaluation using necessary components. In *CVGIP: GRAPHICAL MODELS AND IMAGE PROCESSING*, pages 23–30, Western Australia 6009, 1992.

[23] J.Y. Yang and K. Liu. A new method of extraction continuous curves from images. In *Proceedings of IEEE Asian Electronics Technology Conf.*, pages 244–248, Hong Kong, 1987.

[24] L. O'Gorman and R. Kasturi. *Document Image Analysis.* IEEE Computer Society Press, Los Alamitos, California, 1995.

[25] K. Liu, C.Y. Suen, and C. Nadal. Automatic extraction of items for payment recognition. In *Technical Report, CENPARMI, Concordia University*, Montreal, Canada, 1995.

[26] Joseph Nassif Said. *Automatic Processing of Documents And Bank Cheques.* PhD thesis, Concordia University, Montreal, Canada, 1997.

[27] A. Bleau, J. De Guise, and A. Robert Le Blanc. A new set of fast algorithms for mathematical morphology. In *CVGIP: Image Understanding*, volume 56, pages 178–209, 1992.

[28] J. Serra. *Image Analysis and Mathematical Morphology.* Academic Press, 1982.