International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:4, No:1, 2010

Localizing and Recognizing Integral Pitches of Cheque Document Images

Bremananth R., Veerabadran C. S., and Andy W. H. Khong

Abstract—Automatic reading of handwritten cheque is a computationally complex process and it plays an important role in financial risk management. Machine vision and learning provide a viable solution to this problem. Research effort has mostly been focused on recognizing diverse pitches of cheques and demand drafts with an identical outline. However most of these methods employ template-matching to localize the pitches and such schemes could potentially fail when applied to different types of outline maintained by the bank. In this paper, the so-called outline problem is resolved by a cheque information tree (CIT), which generalizes the localizing method to extract active-region-of-entities. In addition, the weight based density plot (WBDP) is performed to isolate text entities and read complete pitches. Recognition is based on texture features using neural classifiers. Legal amount is subsequently recognized by both texture and perceptual features. A post-processing phase is invoked to detect the incorrect readings by Type-2 grammar using the Turing machine. The performance of the proposed system was evaluated using cheque and demand drafts of 22 different banks. The test data consists of a collection of 1540 leafs obtained from 10 different account holders from each bank. Results show that this approach can easily be deployed without significant design amendments.

Keywords—Cheque reading, Connectivity checking, Text localization, Texture analysis, Turing machine, Signature verification.

I. Introduction

FINANCIAL systems often require a secure way of transaction using bank cheques or demand drafts. The use of these documents increases rapidly because of their ease of negotiation and posted date money trades. Yet, most of the documents are processed by human intervention which in turn causes risk to financial managements. Automatic recognition provides an efficient and accurate solution for document processing without human intrusion. However, localization of integral pitches in a cheque document is a challenging process particularly for systems to possess the ability to read meaningful entities including courtesy amount, legal amount, cheque number, account number, signature, cheque date and bank address. Upon recognizing pitches of cheque number and account number as well as signature, the invoked transaction is performed successfully for rapid trading. To this end, we require a robust scheme for practical implementation in order to cope with challenging issues related to the diverse outlining, mixing of pre-printed/handwritten characters, overlapping of characters and dissimilar backgrounds.

Bremananth R and Andy W. H. Khong are with the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore, e-mail: bremresearch@gmail.com, bremananth@ntu.edu.sg, andykhong@ntu.edu.sg (see http://www.ntu.edu.sg).

Veerabadran C S is with Hexaware Technologies Inc., Jamesburg, NJ 08831, USA, e-mail: vickyitin@gmail.com.

With regards to segmentation, bottom-up and top-down approaches have been proposed. In the former, characters of the entire cheque are localized and subsequently grouped together in order to recognize individual patterns. On the contrary, for a top-down approach, entire pitches are first localized and thereafter characters are extracted, recognized and interpreted. The bottom-up method achieves a poorer performance in terms of reading compared to the latter approach since unless one knows the type of the pitches, it is very challenging to interpret and identify the meaning of the pitches. In general, bottom-up schemes such as that proposed in [1] [2] require 20-30% more computational time to process the entire cheque pattern than the top-down approach. For this reason, we propose a top-down approach to localize entire pitches. Previous works have mainly been focused on exploiting perceptual specific features when reading selective pitches of courtesy amounts, legal amounts, date and signature. Relatively fewer works have been proposed to generalize the cheque recognition process because of its complexity in localizing multiple poses and classes such as machine-printed, hand-printed, free-hand texts and signatures.

A system was proposed for the recognition of the legal amount, courtesy amount and the E13B amount code on Chinese bank cheques [3]. Rafael et al. used a bottom-up approach to locate courtesy amount from Brazilian and USA bank cheques [1] [2]. This recognition was performed using a hybrid neural network. The sliding window method was utilized in extracting signature from an approximate area of the Malaysian, Indian, and Australian bank cheques [4]. A new hybrid approach, which uses an analytical strategy to segment the legal field, was also proposed for reading and verifying the legal amount from Italy cheques [5]. Lee et al. proposed the rule of postprocessing and the cross-referencing algorithms to verify courtesy amount of Korean bank cheques [6]. In [7], an A2iA cheque reader for processing various international cheques was proposed and this provided a unified solution for complete bank cheque processing.

In this paper, we propose a generalized outline solution to recognize different bank cheques using a cheque information tree (CIT) that is based on lexicographical order and weight based density plot (WBDP) algorithms. Texture and perceptual features are exploited for recognizing handwritten pitches while the Turing machine is invoked for interpretation. Our proposed system consists of pre-processing, text entities extraction, segmentation, feature extraction, character recognition and post-processing modules. The Radon transform and bi-cubic interpolation are also employed for orientation estimation and corrections, respectively. Localized fields are

segmented using an eight-way connectivity.

The remainder of this paper is organized as follows. The process of automatic cheque recognition is first described in Section II. Section III deals with the interpretation of cheque information. Experimental results are presented in Section IV while concluding remarks are given in Section IV.

II. Cheque Recognition Algorithm

In this section our proposed method is described. Once a bank cheque image is acquired, it is passed through a series of pre-processing stages, locating active regions by WBDP and fetching coordinates of meaningful entities using the CIT. Normalization is first carried out to enhance the quality of the pitches and to avoid incorrect reading. Gabor texture and perceptual features are then employed for classification. In order to increase the efficiency of legal amounts recognition, a two-stage classifier is employed using Hamming Neural Network (HNN) and Hidden Markov model (HMM). In the post-processing stage, both the Turing machine and Type-2 grammar hierarchy are invoked. The following sections explain the most demanding process of a cheque recognition system (CRS).

A. Aspect Ratios and Binarization

Bank cheques are first acquired by a scanner with 200 dots per inch (dpi) and a 24-bit color depth. A typical cheque aspect ratio can range from 2.10 to 2.50 and as such, the memory requirement for processing the image is dynamically changed. Therefore, to reduce the computational complexity, resizing the acquired image to a fixed dimension is normally required. An aspect ratio of 1.333 is commonly used for image resizing. We require independence between color components due to the non-uniformity of color magnitude, different graphic and object backgrounds as well as different pre-printed and hand-printed foregrounds. Due to the variation of darkness between colors for the background and usage of different color pens, the gray information is also extensively speckled. We propose to employ an optimal thresholding method for both gray scale and binarization conversions. This is based on an approximation of the histogram using the weighted sum of two or more probability densities with normal distribution. The threshold is set as the closest gray-level corresponding to the minimum probability between the maximum of two or more distributions. This results in a minimum error binarization process [8]. To find the background magnitude, the image is divided horizontally. Corner portions are treated as backgrounds and we compute the mean values of both background and foreground pixels. After finding the optimal threshold value, a binarization process is then applied. A set of all 'ON' pixels are used to represent the cheque data. However, since there is no specific statute to fetch the text entities without changing their present meanings, we require a set of rules to trace the text areas and extract local entities precisely. We propose a method of weight based density plot (WBDP) and the CIT where the WBDP draws a straight line over the text areas using weight manipulations. This reveals more density in the text areas rather than non-text areas. The derived coordinate phase is then used to extract peak portions.

B. Weight Based Density Plot

In order to detect areas of the cheque image containing text information, the WBDP is employed to compute weights of the text information. We utilize the fact that the weight of a pixel along an edge increases if the next edge is closer to the previous edge. On the contrary, this weight decreases if the next edge is away from the current edge. Therefore, the change in weight is directly proportional to the distance in terms of the number of pixels between the two edges of an object. It exploits the fact that the edges are denser in the region containing the object than any other region of an image. The edges obtained from the pre-processing phase are used as input to the WBDP. It is useful to note that in some cases where an image may contain unusual artifacts or some other particles partially occluded in the region-of-interest (ROI), finding the accurate Cartesian co-ordinates of the ROI is a challenging problem. In the WBDP, the concentration of edges is determined using the weight assignment scheme given by

$$W_d \propto D \Rightarrow W_d = C_1 D, \tag{1}$$

where W_d and D denote, respectively, the weight decrement factor and the distance between current and previous edge. Here, $0 < C_1 < 1$ is the decay constant. In addition,

$$W_d \propto \gamma \Rightarrow W_d = C_2 \gamma, \tag{2}$$

where γ is a weight increment factor while $0 < C_2 < 1$ is another decay constant. Multiplying (1) and (2), (3) can be derived as

$$W_d^2 = C_1 C_2 D \gamma. \tag{3}$$

Therefore,

$$W_d = \sqrt{C_1 C_2 D \gamma} = \eta \sqrt{\gamma D}, \tag{4}$$

where $\eta = \sqrt{C_1 C_2}$. Using (4), an initial weight value is assigned.

Step 1: Read pixel from the edge detected image. Initialize $W_d = 0$, $D = 0$ and $Flag = 1$.

Step 2: If the current pixel is 'ON' then $D = 0$ and $Flag = 1$.

If $W_d = 0$, then let $W_d = \beta$ corresponding to the starting $\{x, y\}$ coordinates, else if $W_d > 0$ then $W_d = W_d + \gamma$.

Step 3: If the current pixel is 'OFF' and if $W_d > 0$ then $D = D + 1$, $W_d = W_d - \eta\sqrt{\gamma D}$.

If $Flag = 1$ then store the ending $\{x, y\}$ coordinates and $Endwt = W_d$ and $Flag = 0$.

Step 4: If $Flag = 0$, $W_d \leqslant 0$ and $Endwt > W_d$ then set pixels using starting and ending coordinates and assign $W_d = 0$.

Step 5: Repeat **Steps 2 - 4** until entire pixels in the image is scanned.

In above algorithm, the set of 'ON' pixels represents the edge map locations while the set of 'OFF' pixels denotes the edge pixels. This algorithm can be applied for mapping pixel densities of rectangular objects. These objects are scanned as a straight-line map with the process starting from the beginning of every row of the edge map and its weight values being updated with respect to the set of 'ON' and 'OFF' pixels as shown in Fig. 1.

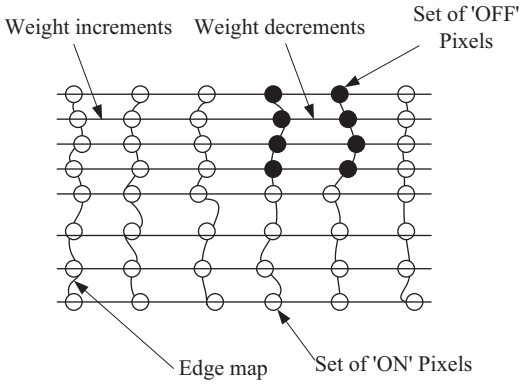

Fig. 1. The process of straight line map using WBDP.

When the first edge pixel 'ON' is found, an initial weight β is assigned to W_d. This weight is then associated with the particular pixel and the next pixel is analyzed. If this subsequent pixel is also an edge pixel, W_d is increased by the weight increment factor γ and this resultant value is in turn allocated to the current pixel. However if the next pixel is 'OFF', then W_d is decremented by $\eta\sqrt{\gamma D}$ as described by Step 3. We note that the degree by which W_d decays depends on both the decay constant η as well as the distance between the current pixel and the most recently encountered edge. As a result, the weight decays at a much faster rate when edges are sparse as compared to regions of dense edges. The procedure is repeated until the entire edge map is processed.

Once WBDP finishes scanning the cheque image, a collection of coordinates is stored in a weight descriptor table (WDT). This table represents regions of the image where the cheque information is available. A method is required to extract the exact and shear locations of regions containing text within the image. The extremity values (top-left and bottom-right) are stored in the entity descriptor table (EDT) for bounding text entities in the cheque. Thus, $x_{\min}$ and $y_{\min}$ coordinates are selected from the WDT for setting up the top-left of the maximum rectangle. In a similar manner, $x_{\max}$ and $y_{\max}$ coordinates are chosen for fitting the bottom-right rectangle as shown in Fig. 2(a). Furthermore, shear pitches may appear in the image where these coordinates are extracted using the maximum fixing rectangle as shown in Fig. 2(b). We ensure that the rectangular regions do not clip over the characters. It also performs a column-wise pixel scan near the left and right ends. If the extreme columns show a presence of higher number of 'OFF' pixels compare to a character region, it implies that the rectangular region has clipped part of a character. In this case, the column-wise scan proceeds in an outward direction until the clipped character has been recovered. We also require a method that confines selected rectangular coordinates which may be overlapped or overwritten. This is due to the slanting nature of pre-printed/handwritten characters or skewness of the text areas present in the image. To resolve these difficulties, EDT is

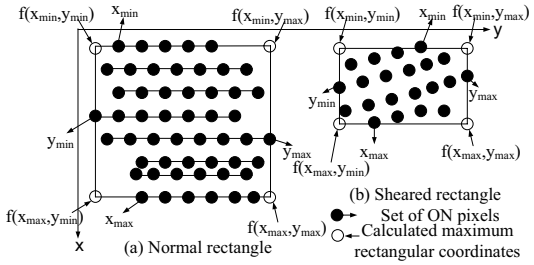

Fig. 2. Illustration of deriving entities using maximum rectangular coordinates approximation (a) rectangle pitches (b) sheared rectangle pitches.

scanned again to find the following constraints. Let R_n and R_m be the set of coordinates of any two rectangles on a cheque image and the following properties are required to satisfy for updating the EDT as

Property 1: If $R_n \subset R_m$, then $(R_m \cap R_n) = R_n$. This property verifies any rectangle coordinates that are completely inside the boundary of rectangle R_m. Therefore, R_n coordinates are eliminated from the EDT and R_m is retained.

Property 2: If $R_n \cap R_m \neq \varnothing$, then $\{R_n,\ R_m\}$ is stored in the EDT, where $\varnothing$ is an empty set.

Property 3: If $R_n \cap R_m = \varnothing$, then, similar to Property 2, $\{R_n,\ R_m\}$ is stored in the EDT.

Property 4: In order to remove straight-line coordinates or non-text areas, coordinates of R_m and R_n are verified with proper rectangle coordinates.

After deriving the pitches, meaningful entities are identified by an outline analysis method. This outline analysis provides a generalized structure for CIT to transverse through the entire pitches in the cheque. The CIT extracts intrinsic coordinates of each entity and forms a tree structure. This form of tree structure is referred to as the check information tree (CIT) shown in Fig. 3. It provides a complete structure to the pitches interpretation and is more efficient to identify the entity types. The CIT is formed based on the outline of the bank cheques and demand drafts (DD). It further extracts each entity and its contours, by identifying contours system which will invoke a specific task for localizing individuals and recognizing them. Pitches are fetched by traversing within the CIT and is based on lexicographical order from which meaningful contours are recognized. In addition, the naming of pitches is carried out based on the abstract syntax notations. Therefore, if a notation A $<$ B then A is lexicographically smaller than B. The CIT transverses across the entire level of the nodes in a structural manner and provides the pitches interpretation. Each Level represents the directives in which pitches are being identified by the system. The notation of CIT traversal is based on the syntax $< Level > . < pitch\ number > . < Bankname >$. A $< NULL >$ symbol represents the system where an appropriate entity is missed or dislocated. This is illustrated as shown in Fig. 4. Based on the above said properties the maximum bounding rectangles have been localized.

The process of extracting the active regions among the other coarse of contour is a challenging process. For that, we used a generalized out-line method to fetch the meaningful

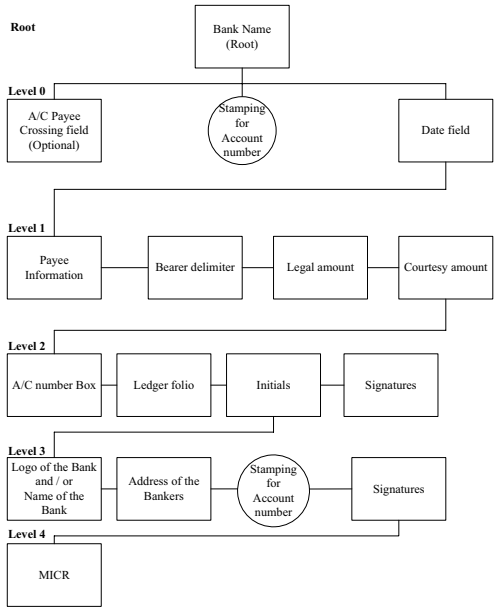

Fig. 3. Outline structure representation using the CIT. *Root* node represents transaction bank and its levels $(0-N)$ indicate significant pitches.

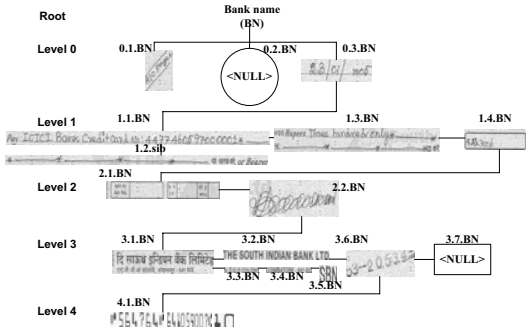

Fig. 4. Representation of lexicographical transverse order for naming each contours.

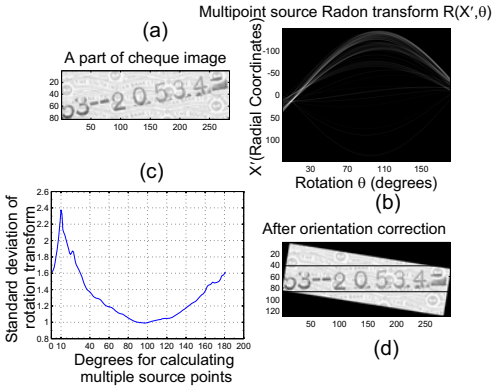

Fig. 5. Rotation estimation and correction process using the Radon transform, a contour of a cheque (a), rotation estimation using multi-point sources (b), standard deviation of the transform (c), rotation correction (d).

Cartesian coordinates. Next, skew angle estimation is invoked for rotation correction to straighten the slant nature of contours. Slant nature of these contours are found based on the average of maximum bounding rectangle. In existing literature, skew detection techniques based on the Hough transform were proposed in [9] [10]. In [11], a boundary growing approach was used to find the skew angle of the document image. Here we employ multiple sensor points using a line integral method based on the Radon transform. This transform employs an orientation estimation to find the skew angle of sheared rectangle pitches. A useful property of this transform is that it determines an internal constitution of an object without changing the interior structure. Multipoint sources are employed to compute the line integrals along parallel beams in a specific direction. A projection of image $f(x,y)$ is a set of line integrals that represents an image while the phase takes multiple parallel-beams from different angles by rotating the source around the center of the image. Multipoint projection are computed for each angle θ. The Radon transform of $f(x,y)$ is calculated using the line integral of parallel paths to the y-axis. The multipoint projection is defined as

$$R(x^{'},\theta)=\int_{-\infty}^{\infty} f(x^{'}\cos\theta-y^{'}\sin\theta, x^{'}\sin\theta+y^{'}\cos\theta)dy^{'} \quad (5)$$

where $R(x^{'},\theta)$ is the Radon transform, $x^{'}$ is the smallest distance to the origin of the coordinate system, θ is the angle of rotation from 0 to π, $x^{'} = x\cos\theta - y\sin\theta$ and $y^{'} = x\sin\theta + y\cos\theta$. The algorithm for estimating the angle of rotation is as follows:

Step 1: An input image as shown in Fig. 5(a) is first rotated counterclockwise using the bi-cubic interpolation method to a specific angle such as 90°.

Step 2: Radon transform is applied by varying θ from 0° to 180°. The maximum projection data and radial coordinates are used to estimate the rotation angle of the image as shown in Fig. 5(b).

Step 3: Standard deviations of the Radon projected data are then computed to estimate the local maximum deviation of the sensor data. This data set is used to compute the maximum rotation angle of the image as shown in Fig. 5(c).

Step 4: The estimated angle $\hat{\theta}$ is used to correct the rotated image to its principal direction. This is carried out by the bi-cubic interpolation, i.e., if $\hat{\theta}$ is positive and less than 90° then the clockwise correction is $-(\hat{\theta}+90^{\circ})$ otherwise if $\hat{\theta}$ is negative and greater than 90° then the clockwise correction is $-(\hat{\theta}-90^{\circ})$. Figure 5(d) shows the rotation correction process after estimating the orientation angle.

C. Vertical Connectivity Checking

We describe the segmentation of individual characters in this section. In existing literature, several methods have been proposed to resolve the problems of character segmentation, i.e., isolating touching characters arising from the slant nature of characters. Integrating segmentation and recognition using

sliding window approach was proposed in [12]. In [13], the entire word was recognized instead of isolating individual characters. We work with vertically connected component analysis using constraint checking which negates any unnecessary characters present in the active area of the cheque. Entire words and individual characters are extracted for legal amount and other cheque entities respectively. During connectivity checking, the size in terms of pixels and also the extremity coordinates of each connection of characters are recorded. The algorithm to achieve the above is as follows:

Step 1: Initialize $(x_{\min}, y_{\min})$ as the size of the image and $(x_{\max}, y_{\max})$ as 0.

Step 2: Find the first 'ON' pixel and store its coordinates (x, y) if it is smaller than $(x_{\min}, y_{\min})$.

Step 3: An eight-neighboring checking module is recursively invoked for ensuring connectivity of 'ON' pixels by including boundary conditions.

Step 4: If the number of connections are above the confined threshold, then fetch that contour as a character object of meaningful entity.

Step 5: Invoke detaching contour module that extracts each entity.

Step 6: Repeat **Steps 3 - 5** until entire levels of CIT objects have been scanned.

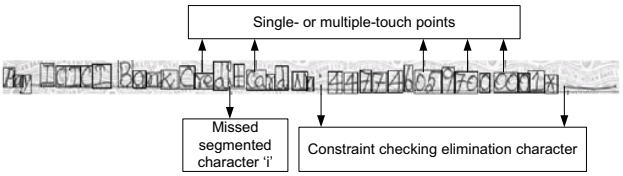

Fig. 6. Representation of touching characters, missing segmented characters and constraint checking used for elimination of unnecessary characters.

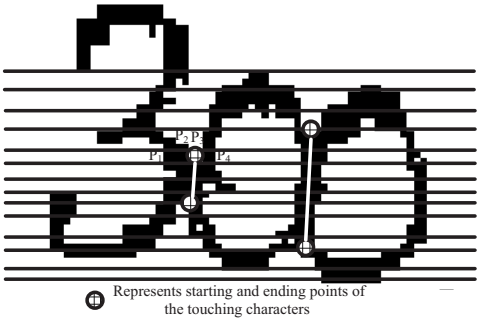

Fig. 7. Representation of touch points detection using scan-line method.

D. Scan-line Touching Point Recovery

After performing connectivity checking, the touch recover module is called. This is invoked by comparing the size of the rectangle corresponding to the touch characters (which often has a larger size compared to foregoing characters) with the average size of the foregoing characters using connectivity checking as shown in Fig. 6. From the large rectangle, diverse types of touch patterns can be identified. This may be in the form of single-touch or multi-touch characters. In the first case, two characters are joined together with a single point whereas for the latter, two characters appear with two or more joining points. The former requires only two points to start and end the dividing paths while on the contrary, the latter requires two or more points to detach the different characters. Courtesy amount, which often consists of a sequence of numerals, may have the problem of continuous single touch characters.

An extended drop-fall and hybrid drop-fall algorithms to separate touching characters were proposed in [2]. In [14], a combination of background and foreground pixels were employed to determine the best path for isolating single- or multiple-touch points of the numeric characters using a mixture of Gaussian probabilities. These methods require a selection of best path to divide the contours. Moreover, in order to increase its performance, the contours should be scanned from a specific point of the touching characters rather than from the beginning of the characters. Hence, we employ a scan line method to find the touch points that is suitable to detach multiple touch points in the same character. It scrutinizes from the left corner of the character image and records the number of crossing points between 'OFF' and 'ON' pixels as shown in Fig. 7. The scan-line algorithm records four flag points P_1, P_2, P_3 and P_4 for non-touching areas of the image and these flags are set as 'True'. If the scanning line passes through the touching portion, then the flags P_2 and P_3 are set as 'False' since these two points are completely hidden by characters '3' and '0'. For this illustrative example, P_1 and P_4 are therefore set to be 'True'. Therefore, the algorithm determines that previous scan line coordinate P_3 is the starting location of the touching character that leads the dividing path. The end points are chosen when P_2 and P_3 are set again as 'True'. The process terminates at end-point coordinate P_3 and the dividing method replaces the set of black pixels by white pixels that lie on the touching path in order to discriminate the touching contours. The constraints of the algorithm are as follows:

Constraint 1: If (P_1 = 1, P_2 = 1, P_3 = 1 and P_4 = 1) then non-touching part is identified by $1 \bullet 1 \bullet 1 \bullet 1 = 1$, where $\bullet$ represents 'AND' operation executed on a Boolean data type.

Constraint 2: If (P_1 = 1, P_4 = 1) and (P_2 = 0, P_3 = 0) then the touching part is determined by $1 \bullet 0 \bullet 0 \bullet 1 = 0$.

Once the touching location is found, the dividing path starts from the coordinate of the previous crossing scan line to the final ending scan line coordinate.

E. Pitches Normalization

The extracted characters are first normalized before features are being extracted. Three techniques have been carried out to normalize the pitches. They are slant correction, resizing and thinning.

1) Slant Correction: The handwritten and preprinted characters or graphics entities are often overlapped because of its writing or printing style. This problem can be overcome by a

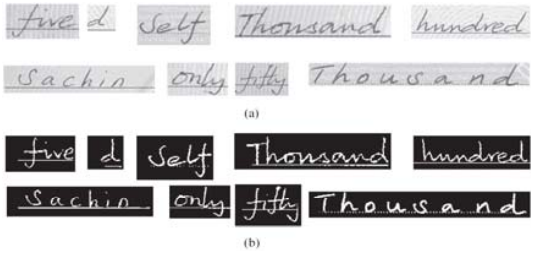

Fig. 8. Representation of slant correction process (a) slanting nature handwritten words and characters (b) result of slant correction process.

series of shear transformations. We utilize the fact that a word has maximum width when it is slanted and minimum width when vertical. We first find the maximum slant angle θ of the character in a given image by computing

$$\begin{aligned} R_d &= y_{\max} - y_{\min}, \\ H_1 &= C_o/R_d, \end{aligned}$$

where C_o is the number of 'ON' pixels in a row of the character pitch. If $H_1 = 1$, then we define the sum of a series of slant transformation by

$$S_l = S_l + C_o^2. \tag{6}$$

Finally the maximum slant angle is used for slant correction. We perform a series of shear transformation using the following procedure:

Step 1: Initialize $M = Width$, $N = Height$, $x = 0$ and $y = 0$.

Step 2: Compute $i = (x + (y \times \tan\theta))$, $j = y$, $V = P_{x,y}$ and $S_{i,j} = V$.

Step 3: Increment i and j by 1 and repeat the above steps until $x = M$ and $y = N$. The results of this algorithm is shown in Figs. 8(a-b).

2) Thinning and Thinking: After correcting the slant angle for each of the characters, they are converted into a fixed size contour of 60 x 40 using bilinear transformation. The process of thinning is another important task that transforms a contour of the segmented characters to a set of topological objects [15]. It defines the erasure of black pixels such that an object without holes erodes to a minimally connected stroke located equidistant from its nearest outer boundaries. On the contrary, an object with holes erodes to a minimally connected ring mid-way between each hole and it's nearest outer boundary. After passing through the thinning process successively, a character of any pixel value 'ON' and at least one 8-way neighboring value 'OFF' is identified. If the number of 'ON' pixels for P_1 ranges from 2 to 6 and the total number of 'ON' to 'OFF' transitions is equal to 1, $(P_2 \bullet P_4 \bullet P_6) = 0$ and $(P_4 \bullet P_6 \bullet P_8) = 0$, then pixels ranging from P_1 to bordering pixels in the contour are deleted. However, if the number of 'ON' pixels of P_1 ranges from 2 to 6 and the total number of 'ON' to 'OFF' transitions is equal to 1, $(P_2 \bullet P_4 \bullet P_8) = 0$ and $(P_2 \bullet P_6 \bullet P_8) = 0$, then point P_1 is flagged for deletion while the border area of pixels in the contour are not deleted. The above process is repeated until no further points are deleted.

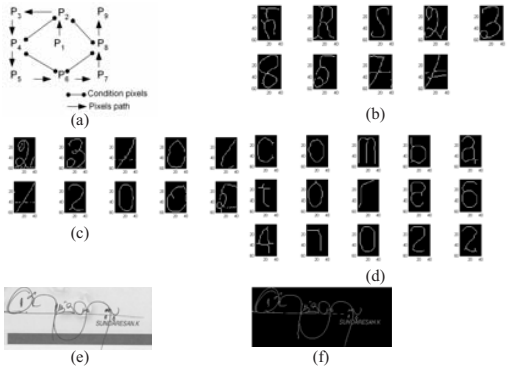

Fig. 9. Representation of thinning process and its results (a) Pixel mining in the thinning process. Visualization of thinning results (b) courtesy amount (c) Date field (d) Bank address (e) signature image inputs to the thinning algorithm (f) removal of graphics objects and its output.

This is shown in Figs. 9(a-d). Thinning can also be applied to the background of an image. The thinned strokes that generate the original objects are minimally connected and therefore the background of the separating strokes is connected throughout the image. During the connectivity checking process, due to the connectivity of pixels, signature pattern could be overlapped with graphical object which is preprinted in the cheque. This may lead to the appreciation of unwanted information or depreciation of useful information. This appreciation and depreciation can be resolved by our algorithm as shown in Figs. 9(e-f). After thinning, the entire contours will then be processed by the thickening method in order to acquire fixed patterns of characters. This can be done by adding more pixels to the exterior patterns based on 8-way connected components.

F. Feature Extraction and Recognition

This phase plays a vital role in recognizing numerals, characters and signatures present in the cheque image. Recognizing entire cheque pitches is a challenging process when a single classifier such as the Radial Basis Function (RBF) or Back Propagation Network (BPN) is used without any special features. Here, we propose to use combined classifiers comprising of Gabor feature Hamming neural network classifier (HNN) and a holistic feature hidden Markov model (HMM). The former is employed to recognize entities such as date, payee, bank addresses, cheque number, courtesy amount and other information while the latter is performed on legal amount pitch using two-stage classifiers. Initially the legal amount is validated by Gabor feature HNN and then confirmed using a holistic feature HMM.

Signature is first authenticated against the account number using Gabor features with the Euclidean norm distance measure (ENDM). Handwritten character recognition is mainly based on biological findings of the mind and vision systems. It motivates the multi-channel Gabor filters such that they provide psychophysical results of the visual information in a quasi-autonomous manner. The Gabor function $g(x, y)$ is an isotropic 2-D Gaussian function. Defining δ_x and δ_y as

the scaling parameters of the Gaussian envelope along the x and y axes, respectively [16], f as the radial frequency of the sinusoid and θ as the orientation of the Gabor filter, the performance of the Gabor filter depends on choosing its parameters $\{\theta, f, \delta_x, \delta_y\}$. We note that the Gabor filter produces responses that varies with parameter settings. Here we employ Gabor optimization such that its parameters are efficiently used. This in turn leads to a non-redundant feature extraction. We first choose the parameters of Gabor filters that provide strong responses to the recognition process. Since it is applied for character feature extraction, a strong deviation of signals between divergent classes of characters are chosen. In addition, ambiguity of characters is also considered in the feature extraction. Thus, the objective is to build a set of Gabor filters that is capable of generating efficient features in the presence of different artifacts.

In most cases, relevant features are unidentified and a large number of features are extracted to better recognize a particular character. However, without explicitly employing a feature selection strategy, many of these features could be redundant or even irrelevant to the classification task. Identifying which features to utilize in a classification task is referred to as feature selection.

Feature selection involves the use of either the filter bank or filter design approach. Although filter bank approach was proposed in order to choose Gabor parameters in an improvised manner [17] [18] no optimal solution is provided for a particular task. Another problem is the redundancy of convolutions among the filters which, in turn, require more computation during feature extraction. In order to address these issues, a few sets of filters are designed to classify the given patterns efficiently using filter design approach. This in turn provide an optimal set of Gabor filters which, as a consequence, reduces the computational complexity compared to the filter bank approach. Selection of the set of Gabor filters for a specific problem domain is associated with optimizing the Gabor parameters efficiently. As an illustrative example, suppose a character recognition requires G number of filters such that $4G$ parameters is required to be optimized. Therefore, if $G = 20$ filters are needed for the application, then 80 parameters are required to be optimized. In general, optimizing such a large dimensional problem is challenging and demands high computational complexity in order to achieve optimal states.

In the context of Gabor filter optimization and selection, the Boltzman learning based stochastic optimization combined with k-means clustering approach is employed for filter design and selection process. The parameters of the Gabor filters are first optimized by the Boltzman machine according to the response of the filters while the k-means clustering algorithm is used to group the redundant filters present in the optimization process. A common challenge is the recognition of ambiguous characters such as 'O/D' or '5/S' or 'B/8' or 'A/4' which have similar characteristics in both holistic and perceptual features. This in turn leads the system to ambiguous states. To resolve this issue, an optimized filter design approach is used to extract features in order to discriminate characters analytically. It is useful to note that we used texture-based analysis which is invariant to any pattern variation.

Training is based on a neural network which accepts bipolar features. We employ a recurrent network consisting of N two-state units. These states can be chosen from the bipolar space, i.e., each Gabor filter parameters are converted to a bipolar string. If G filters are necessary for the filter design and parameters require M bipolar string, then GM length of patterns are needed. Each of the four parameters are determined using $M/4$ bipolars. If, for example, a character contour of size 60×40 is sub-divided horizontally into three sets of 20×40 size sub-patterns, then a total of 20 Gabor kernels are used to convolve each sub-patterns. The combination of mean and standard deviation values of each character strip have been calculated as $\langle\{{}^1_1\mu^1_1\sigma,\ {}^2_1\mu^2_1\sigma,\ \cdots,\ {}^{20}_1\mu^{20}_1\sigma\}, \{{}^1_2\mu^1_2\sigma,\ {}^2_2\mu^2_2\sigma,\ \cdots,\ {}^{20}_2\mu^{20}_2\sigma\}$ $\{{}^1_3\mu^1_3\sigma,\ {}^2_3\mu^2_3\sigma,\ \cdots,\ {}^{20}_3\mu^{20}_3\sigma\}\rangle$, where the superscript denotes the feature index while the subscript represents strip index. Therefore, 120 features are extracted from each character image. The HNN consists of two layers and it decides which test set is nearer to the trained set. The first layer is the feed forward layer (FFL) which is used to calculate a maximum score of the input patterns while a second recurrent layer (RL) is used to select the maximum score among the input character patterns. Each neuron in the FFL is set up to give a maximum response to one of the trained patterns. If the test set is the same as the trained set, the recurrent network is allocated a maximum score. The weight initialization process of the HNN is given by $w_{ij} = x_{ij}/2$, $w_{i0} = n/2$, where w_{ij} and x_{ij} are the weight coefficient and input features, w_{i0} is a bias value and n is the number of features from the character. In HNN, the number of neurons in the FFL is the same as the number of neurons in the recurrent layer. When a test feature is given to the FFL, the output from each of the neurons in the FFL determines the Hamming distance from the character in the training set. This Hamming distance between two character patterns is a measure of the error between the character patterns. Therefore, if the entire values are changed in the character patterns, the neuron that corresponds to that pattern is 0. The function of the RL is then to select the neuron with the maximum output. The final output of the RL contains a larger positive value at the neuron that corresponds to the nearest character pattern while all other neurons produce a 0 value. In the recognition process, the output is based on the value of the FFL. The initial outputs of the RL is equal to the score produced by the FFL and after several iterations, all of the outputs will converge to zero except for those recognized characters. The testing process of HNN is terminated if no change is detected between iterations.

Recognizing the legal amount is based on the holistic approach. The HMM-based handwritten classifier was proposed in [19] and a survey of the holistic paradigm was discussed in [20]. However, the holistic performance is degraded proportionally to the number of lexicon trained by the network. This is due to the assignment of a probability variable to several lexicons recognized by the classifier. Moreover, legal amount pitches are a combination of uppercase and lowercase handwritten letters, hand-print or cursive characters. From these amalgamations, assignment of probabilities according to the perceptual feature is insufficient for the classifier to

discriminate between them properly. Furthermore, a single classifier result is not an exclusive decision when it comes to classifying cursive words with widely varied patterns. Therefore, we use a combination of perceptual and Gabor features to discriminate the legal amounts. It should be noted that, in general, holistic recognition is more efficient than analytical approach. The network is trained with different combinations of both uppercase and lowercase legal amounts. The minimum error rate of both classifiers is then used for testing. The ascender and descender features are taken as perceptual features and the body of a word is taken as the Gabor feature. The body of the text is resized and convolved with the Gabor kernels. The combination of these two features are trained and tested by the HMM.

G. Signature Verification

In [21], a combination of two-stage neural network and radial basis function was used to recognize signature in an off-line mode. An approximate area estimation method using a sliding window to extract signature from the cheque leafs was proposed [4]. Signature confirmation phase should verify the already existing features of the signature against the newly extracted features. We utilize Gabor filters to extract features from the signature pattern but the number of features and classifier design varies since a signature is an entity that determines the genuinity of patterns signed by the subjects. To address this variation, each subject must enroll five different signatures to assist the confirmation phase. This multiple enrolment is to facilitate learning of stroke variation generated by the same subject under different occasions. From these image patterns, a mixture of a feature set is constructed. This set consists of 60 features derived from the extracted signature contour of size 80×120 which, in turn, is sub-divided vertically into three sub-patterns each of size 80×40. From each sub-pattern, 20 features $\langle \{ {}^{1}_{1}\mu,\ {}^{2}_{1}\mu,\ \cdots,\ {}^{20}_{1}\mu \} \rangle$ are extracted. The weighted distance maintained in the system ranges from 0.0 to 0.4. If this weighted distance is greater than 0.4, then the system decides that the signature pattern is not from a genuine source. The mean value of each character strip is computed by changing the frequency and orientation of the kernels.

III. Interpretation

An interpretation phase is proposed to syntactically verify the recognition errors and facilitate the identification of writing errors, simultaneously. In [22], the syntax analysis phase is proposed to verify the recognition of numerals in a French cheque. Courtesy amount syntax checking using the deterministic finite automata after recognizing the individual numerals was also proposed [1]. In our approach, Type-2 grammar is used to check the syntactic context. The grammar is stated as follows: Let G_a be the Type-2 grammar consisting of $(\{< S >, < A >, < B >, < D >, < PU >, < M >\}, \{0|1|2|, \cdots, |9, /| - |\_|., JAN|FEB|, \cdots, |DEC\}, < S >, P)$, where $< S >, < A >, < B >, < D >, < PU >, < M >$ represent a set of non-terminals while numbers, punctuation marks and list of months denote a set of terminals. In addition, $< S >$ denote a distinguished element of non-terminal called starting symbol and P represents a finite non-empty set of productions. The set of production rules of date pitches in Backus-Norm form are given as follow:

$$
\begin{array}{lll}
< S > & ::= & < A >< D > | < D >< PU >< S >, \\
< A > & ::= & < D >< PU >< A > | < D >< D > \\
 & & < PU >< A >< M >< B > | < D > \\
 & & | < D >< D >< D >, \\
< B > & ::= & < PU >< D > | < PU >< D > \\
 & & < D >< D >, \\
< D > & ::= & 0 \mid 1 \mid 2 \mid, \cdots, \mid 9, \\
< PU > & ::= & / \mid - \mid \_ \mid., \\
< M > & ::= & JAN \mid FEB \mid, \cdots, \mid DEC.
\end{array}
$$

The courtesy amounts are crosschecked with legal amounts. This is achieved by parsing the courtesy amount field after the legal grammar is confirmed. The cross-error checking engine will produce an error if any mismatch exists. A two-way Turing machine (TM) is used to recognize the legal and courtesy amounts. The output of these two phases are then fed to the cross-error checking engine in order to verify both combinations.

This two-way machine is defined by $TM = (Q, I, Z, f, q_o, B, F)$, where Q is a finite set of states, Z is a finite set of acceptable symbols and $Q \cap Z = \varnothing$, B is a symbol of Z which represents a blank, I is a set of input symbols, f is a function which defines the mapping $QXZ \rightarrow QXZ\ X\{Left, Right\}$, q_o is a starting state and F is a set of stop states. Figure 10 illustrates the courtesy amount Turing machine parser.

It is important to note that alphabets $< AP >$ and q_o pass blanks or alphabets such as currency signs in front of the courtesy amount. This is depicted by the label '$< AP >, B : R$' which returns the TM to state q_o and moves the control of TM head to the right of the processing string until the blanks or alphabets are exhausted. Whenever any digit ($< D >= 0|1|2| \cdots |9$) is found, the control advances to state q_1. This is denoted by the label '$< D >: R$'. The TM remains in state q_1 until it reads '$< D >$' again and it moves towards the right of the processing string. If any blank occurs then the control advances to state q_3. Alternatively, if the decimal point '$< DB >: R$' is found, then the control moves towards the right and advances to state q_2. It stays in state q_2 as long as the TM reads '$< D >$'. Whenever TM scans a delimiter '$< T >$' or a blank, the control advances to state q_3 and is represented by the label '$< T >, < B >: R$'. If it encounters the '$< T >$' in state q_3, the control proceeds to state q_4 and if this state comes across as a blank, it then enters state q_5 and the process is terminated. Likewise, from state q_3, a blank may be encountered and, under such a condition, the control enters into a halt state. The advantage of using TM is that it accepts all grammars and the head of the TM can aptly move back and forth between the sentence in the form of machine learning.

IV. Experiments on Cheque database

The overall cheque recognition accuracy depends on the entire process as discussed above. Our method was verified with bank cheques and demand drafts. A database of cheque

TABLE I
RECOGNITION ACCURACY OF CHEQUES.

Phases	Total No. of Training Samples	Recognized Samples		Training set%		Testing set%	
		Train	Test	Accept	Reject	Accept	Reject
1	440	424	598	96.36	3.64	90.60	9.4
2	660	648	402	98.18	1.82	91.36	8.64

TABLE II
RECOGNITION ACCURACY OF DEMAND DRAFTS.

Phases	Total No. of Training Samples	Recognized Samples		Training set%		Testing set%	
		Train	Test	Accept	Reject	Accept	Reject
1	132	128	293	96.96	3.04	95.12	4.88
2	110	105	302	95.45	4.55	91.51	8.49

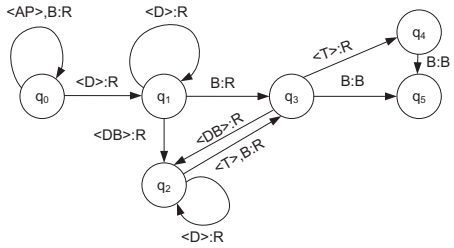

Fig. 10. Turing machine for courtesy amount interpretation.

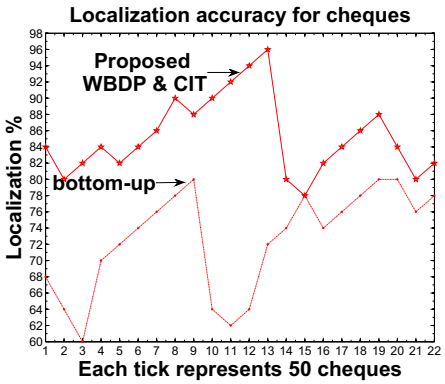

Fig. 11. Localization comparison of bottom-up and WBDP & CIT.

leafs consists of 1540 images derived from 1100 cheques and 440 demand drafts were used to verify the accuracy. The cheque database was collected from 22 banks. This database is obtained from ten different subjects each with five different hand-written cheques. Twenty hand-written demand drafts were collected from each bank.

A. Localization and Vertical Connected Component

Using the vertical connected component, unwanted symbols such as a period, comma and other special symbols were successfully eliminated using the process of constraint checking. Our localization method has been compared with the bottom-up approach by arranging, from high to low, magnitudes of color in both writing and backgrounds of bank cheques. Due to the large variations in writing and backgrounds, an average of 27.36% false localization is achieved by the bottom-up method while only 14.73% false alarm is achieved by WBDP as shown in Fig. 11. We note that the WDBP provide better localization when the decay constant η, weight increment and decrement factors are properly chosen.

B. Accuracy in Date Pitches

Verification of dates was evaluated with different formats of date written by the subjects. Errors in the date field arise due to, for example, any correction made by a patron. In addition, a patron may erase pre-printed date pitches or write

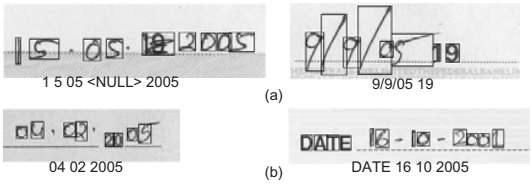

Fig. 12. Date field recognition process (a) error in verification (b) correctly verified.

the transaction date before or after these pre-printed pitches. Among 1540 leafs, date field corresponding to 1442 leafs were recognized precisely but the remaining 98 leafs were wrongly identified due to variation of date format, date rubout, poor writing amongst others. Figures 12(a-b) show typical recognition failure and success.

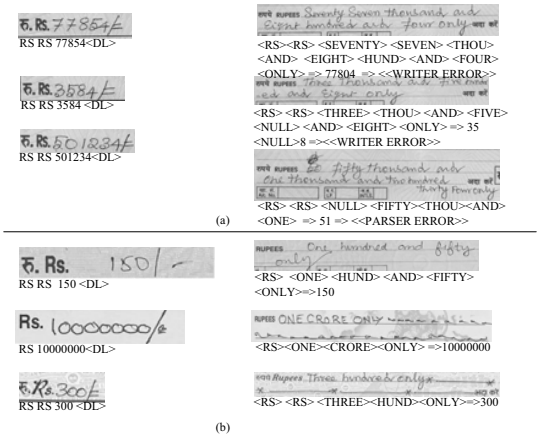

Fig. 13. Representation of legal and courtesy amount crosschecking process (a) incorrect reading produced either by the parser or writers (b) exact matching of both fields.

Fig. 14. A typical transaction of the proposed system by recognizing signature, cheque number and courtesy amount.

C. Courtesy and Legal Pitches

Courtesy amount is crosschecked with legal amount in order to obtain an exact mapping of both fields. The Turing machine crosschecking engine assists the interpretation process to verify these fields. If any field causes error, then it should be verified by the error verification module. The recognition module of both entities is trained and desired patterns are tested. Some of the outcomes are illustrated as shown in Fig. 13. For courtesy amount, '$< DL >$' or '$< NULL >$' strings are used to denote delimiters which, in turn, are for terminating the Turing machine verification process. The delimiter for legal amount is '$< ONLY >$'. Among 1540 leafs, both fields were recognized precisely in 1400 leafs because of different type of writing styles. Leafs with conflicting amount were used for processing. To test the system, 70 leafs were collected with different courtesy and legal amounts. The entire leafs were perfectly negated and passed for "rejection for modification" by the TM cross-error checking engine.

D. Accuracy in Signature

Accuracy of signature verification is evaluated based on 1100 signatures extracted from the cheque leafs. The system was tested in two different phases. In the first phase, signature features were compared with the remaining feature sets. Defining n as the number of feature set, it requires $n(n-1)$ possible comparisons in order to recognize the entire subjects. The genuine and false subject features were matched with respect to the weighted distance of 0.0-0.4 and a non-match status is given when this distance is above 0.4. Defining μ and σ as the mean and standard deviations of genuine signature features respectively, the degree-of-freedom $\nu = \mu(1-\mu)/\sigma^2$ = 395.35 was achieved. In the second phase of the experiment, the account number field was used as a principal key to verify features of the signatures. This second experiment was performed in the form of a one-to-one matching. If the weighted distance of the comparison process was less than or equal to 0.4, the test was accepted, else it was rejected.

The following metrics of signature verifications were performed: Genuine accept rate (GAR): If the system was validated with the genuine subjects and the results were also positive then it is referred as GAR. To verify the GAR, 440 subjects were trained and tested against the genuine subjects. The GAR obtained in our experiment was 96.5%. False rejection rate (FRR): If the system was confirmed with genuine subjects but the responses were negative, it is referred to as FRR. In our experiments, it was found that the FRR was 3.4%. True rejection rate (TRR): If the system was verified with false subjects and outcomes were also negative, then it is known as TRR. Our experiments yielded 426 rejections out of 440 subjects. False positives rate (FPR): If the false subjects were verified with the system and reactions were positive, then it is referred as FPR. Our experiments produced 3.18% of FPR.

E. Overall Accuracy

Recognition of entire fields was verified in two different phases. During first phase, 440 cheques and 132 demand drafts were trained and verified against the remaining 660 and 308 test sets, respectively. In phase two, the system was trained using 770 samples, (inclusive of 660 cheques and 110 demand drafts) i.e., from each subject, three cheques were chosen and for demand draft from each bank, five leafs were selected for training. The remaining 440 cheques and 330 demand drafts were treated as test samples. Tables I and II show the accuracy of cheque and demand drafts respectively. From the result we can observe that the recognition accuracy was proportionate to the number of samples that we trained and tested. The average recognition accuracy for cheques in both testing and training sets were 94.12% and 94.76%. Rejection rate was 5.87% and 5.24% for cheque and demand draft samples, respectively. Figure 14 shows a sample cheque processing by identifying account holder's signature, cheque number and courtesy amount.

V. Conclusions

We propose an approach for cheque image recognition. We employ the WBDP to localize text in an efficient way. This algorithm reduces the computational complexity of fetching meaningful pitches as well. Independency of cheque outline was fixed by the lexicographical order based CIT which gave a generalized solution for transversing through the entire pitches of the cheque. Segmentation and touch-point recovery were also carried out by connected component analysis and scan line methods respectively. In addition, slant correction of cheque characters was performed effectively. Texture and perceptual features were utilized to recognize different set of characters. These features provided considerably high accuracy compared to that when using perceptual features only. The crosschecking process performed better in terms of detecting mistakes for both legal and courtesy amounts. Signature verification classifier was also employed for cheque authentication process. Recognition rates of the suggested methods were found to be better for practical deployment in bank transactions.

References

[1] R. Palacios, A. Gupta, and P. S. Wang, "Handwritten bank cheque recognition of courtesy amounts," *Int. Journal of Image and Graphics*, vol. 4, no. 2, pp. 1–20, 2004.

[2] ——, "Feedback-based architecture for reading courtesy amounts on cheques," *Journal of Electronic Imaging*, vol. 12, no. 1, pp. 194–202, 2003.

[3] D. Wang, "IFCRS : An information fusion based check recognition system," in *Proc. GCIS, 2009 WRI Global Congress on Intelligent Systems*, vol. 2, 2009, pp. 243–247.

[4] V. K. Madasu, M. Hafizuddin, M. Yusof, M. Hanmandlu, and K. Kurt, "Automatic extraction of signatures from bank cheques and other documents," in *Proc. VIIth Digital Image Computing: Techniques and Applications*, 2003, pp. 591–600.

[5] V. D. Lecce, G. Dimauro, A. Guerriero, S. Impedovo, G. Pirlo, and A. Salzo, "A new hybrid approach for legal amount recognition," in *Proc. 7th Int. Workshop on Frontiers in Handwriting Recognition*, Amsterdam, 2000, pp. 199–208.

[6] T. C. Lee, E. J. Kim, and Y. Lee, "Error correction of Korean courtesy amounts in bank slips using rule information and cross-referencing," in *Proc. Int. Conf. on Document Analysis and Recognition*, 1999, pp. 95–98.

[7] N. Gorski, V. Animov, E. Augustin, O. Baret, D. Price, and J. C. Simson, "A2iA cheque reader: A family of bank recognition systems," in *Proc. 5th Int. Conf. on Document Analysis and Recognition*, 1999, pp. 523–526.

[8] S. Milan, H. Vaclav, and B. Roger, *Image Processing Analysis and Machine Vision*. ITP, 1999.

[9] S. N. Srihari and V. Govindaraju, "Analysis of textual images using the Hough transform," *Machine Vision Application*, vol. 2, pp. 141–153, 1989.

[10] S. C. Hinds, J. L. Fisher, and D. P. Amato, "A document skew detection method using run-length encoding and Hough transform," in *Proc. Int. Conf. on Pattern Recognition*, 1990, pp. 464–468.

[11] P. Shivakumara, G. H. Kumar, D. S. Guru, and P. Nagabhushan, "A novel technique for estimation of skew in binary text document images based on linear regression analysis," *Sadhana*, vol. 30, no. 1, pp. 69–85, 2005.

[12] G. Martin, R. Mosfeq, and J. Pittman, "Intergrated segmented and recognition through exhaustive scans learned saccadic jumps," *Int. Journal of Pattern Recognition and Artificial Intelligence*, vol. 3, pp. 831–847, 1993.

[13] J. H. Kim, K. K. Kim, C. P. Nadal, and C. Y. Suen, "A methodology of combining HMM and MLP classifier for cursive word recognition," in *Proc. Int. Conf. Pattern Recognition*, vol. 2, Spain, 2000, pp. 319–322.

[14] Y.-K. Chen and J.-F. Wang, "Segmentation of single or multiple-touching handwritten numeral string using background and foreground analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1304–1317, 2000.

[15] V. M. Nagendraprasad, P. S. P. Wang, and A. Gupta, "Algorithms for thinning and rethickening binary digital patterns," *Digital Signal Processing*, vol. 2, pp. 97–102, 1993.

[16] Y. Hamamoto, S. Uchimura, M. Watanabe, T. Yasuda, Y. Mitani, and S. Tomota, "A Gabor filters-based method for recognizing handwritten numberals," *Pattern Recognition*, vol. 31, no. 4, pp. 395–400, 1998.

[17] K. J. Anil, N. K. Ratha, and S. Lakshmanan, "Object detection using Gabor filters," *Pattern Recognition*, vol. 30, no. 2, pp. 295–309, 1997.

[18] K. R. Namuduri, R. Mehrotra, and N. Ranganathan, "Efficient computation of Gabor filter based multiresolution responses," *Pattern Recognition*, vol. 27, pp. 925–938, 1994.

[19] A. Vinciarelli, S. Bengio, and H. Bunke, "Offline recognition of unconstrained handwritten texts using HMMs and statistical language models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 709–719, 2004.

[20] S. Madhvanath and V. Govindaraju, "The role of holistic paradigms in handwritten word recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 149–164, 2001.

[21] H. Baltzakis and N. Papamarkos, "A new signature verification technique based on a two-stage neural network classifier," *Engineering Applications of Artificial Intelligence*, vol. 14, pp. 95–103, 2001.

[22] L. Heutte, P. Barbosa-Pereira, O. Bougeois, J. V. Moreau, B. Plessis, P. Coutellemont, and Y. Lecourtier, "Multi-bank cheque recognition system: consideration on the numeral amount recognition module," *Int. Journal of Pattern Recognition and Artificial Intelligence*, vol. 11, no. 4, pp. 595–618, 1997.

Bremananth R received the B.Sc and M.Sc. degrees in Computer Science from in 1991 and 1993, respectively. He obtained M.Phil. degree in Computer Science and Engineering in 2002. He received his Ph.D. degree in 2008 from Department of Computer Science and Engineering, PSG College of Technology, Anna University, Chennai, India. Presently, he is working as a Research Fellow, at Nanyang Technological University, Singapore. He received the M N Saha Memorial award for the best application oriented paper in 2006 by Institute of Electronics and Telecommunication Engineers (IETE). His fields of research are acoustic imaging, computer vision, biometrics and multimedia.

Veerabadran C S received his Diploma in Information Technology in 2001 from Nachimuthu Polytechnic, Coimbatore, India and his B.E. degree in Computer science and Engineering in 2005. He has 5+ experience in software development. Presently, he is working as a senior software engineer at Hexaware Technologies Inc. Jamesburg, NJ, USA. He has over five years of experience in Software Design and Development in the field of Air Cargo Transportation & Logistics. His research interests include image processing, computer vision, software engineering and other related fields in computer science.

Andy W. H. Khong received the B.Eng. degree from Nanyang Technological University, Singapore, in 2002 and the Ph.D. degree from Department of Electrical and Electronic Engineering, Imperial College London, London, U.K., in 2005. He is currently an Assistant Professor in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Prior to that, he was a Research Associate (2005-2008) in the Department of Electrical and Electronic Engineering, Imperial College London. His postdoctoral research involved in developing signal processing algorithms for vehicle destination inference as well as the design and implementation of acoustic array and seismic fusion algorithms for perimeter security systems. His research was mainly on partial-update and selective-tap adaptive algorithms with applications to mono- and multichannel acoustic echo cancelation for hands-free telephony. He has also published works on acoustic blind channel identification for speech dereverberation. His other research interests include speech enhancement and blind deconvolution algorithms.