Pertanika J. Sci. & Technol. **11(2)**: 157 - 171 (2003)

ISSN: 0128-7680

Single Slice Grouping Mechanism for Recognition of Cursive Handwritten Courtesy Amounts of Malaysian Bank Cheques

Md. Nasir Sulaiman & Marzuki Khalid*

Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia

**Centre for Artificial Intelligence and Robotics (CAIRO), Universiti Teknologi Malaysia, Jalan Semarak, 54100 Kuala Lumpur, Malaysia*

E-mail: nasir@fsktm.upm.edu.my

Received: 24 April 2001

ABSTRAK

Mekanisme mengumpul hiris tunggal untuk pengecaman melibatkan proses memotong secara menegak ke atas imej hiris demi hiris, mengumpul setiap hirisan mengikut lebar tertentu dan kemudiannya diuji untuk dicamkan menggunakan Rangkaian Neural yang sudah terlatih. Imej mengandungi jumlah berangka tulisan tangan dan bersambung yang diperoleh daripada cek-cek bank di Malaysia. Seni bina Rangkaian Neural tiga paras dengan fungsi ralat baru untuk algoritma pembelajaran *Backprogation* digunakan. Pendekatan ini menghasilkan keputusan yang baik untuk mengecam jumlah yang mengandungi dua atau lebih angka-angka yang bercantum secara sambung atau secara sentuh.

ABSTRACT

Mechanism to group single slice for recognition involves the process of cutting vertically across an image slice by slice, group every slice at a certain width and tested for recognition using a trained Neural Network. The image contains cursive handwritten courtesy Amounts of Malaysian bank cheques. A three layer Neural Network architecture with the new error function of Backpropagation learning algorithm is used. This approach yields good recognition results with faster convergence rates.

Keywords: Cursive handwritten courtesy amount, Malaysian bank cheques, recognition, neural networks, backpropagation learning algorithm, error function

INTRODUCTION

Automated bank cheque processing is an active research area involving offline cursive hand written recognition (Knerr *et al.* 1998; Dimauro *et al.* 1997). Due to the different types of cheques and languages, it is rather difficult to use any commercial recognition software to solve bank cheques of a particular country. In Malaysia, bank cheque processing is done manually. To automate such a process in Malaysia requires a huge amount of resources as well as extremely good research. This is due to the fact that Malaysian bank cheques are usually written in two different languages, English and Malay.

At the Center for Artificial Intelligence and Robotics (CAIRO, UTM), research in automated processing of Malaysian bank cheques is currently being carried out. Three different strategies are employed:

1. to automatically find the location of the date, the courtesy amount and the legal amount of a bank cheque,
2. to segment and recognise the legal amount, and
3. to segment and recognise the courtesy amount and date.

In this paper, we discuss the technique of recognising the courtesy amount of Malaysian bank cheques by single slice grouping and recognising it. The pre-processing technique is presented first before the recognition phase. It can be divided into 4 processes: scanning, filtering or removing noise, thresholding and thinning. The effectiveness of this technique is tested on several hundred Malaysian bank cheques from a variety of banks.

EXAMPLE OF THE HAND WRITTEN COURTESY AMOUNT

The courtesy amount is a sequence of handwritten or typewritten digits which may include RM (Ringgit Malaysia), comma, decimal or slash to represent the amount of money written on any Malaysian bank cheques. The styles of handwriting can be categorised as follows:

- unconnected digits.
- joint digits.
- touched digits.
- overlapped digits.

Fig. 1 shows some examples of the styles of handwriting of the courtesy amounts obtained from Malaysian bank cheques.

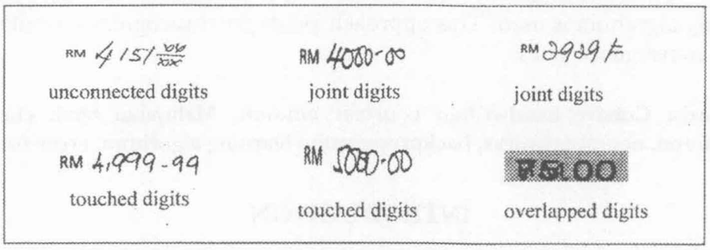


Fig. 1: Examples of the handwritten courtesy amount of Malaysian bank cheques

Due to the large variation of bank cheques in Malaysia which resulted in many levels of difficulty when the images are processed, this research involves only bank cheques that are based on the following assumptions:

- The objects of the image are located and scanned manually.
- The objects of the image are within the box.
- The objects of the image are not mixed with other images such as marking, stamping, etc.

SYSTEM OVERVIEW

The process to recognise off-line cursive handwritten courtesy amount is divided into 2 phases: Pre-processing phase and Recognition phase. The overall process can be illustrated as in *Fig. 2.*

After the recognition phase, the output is expected to contain a list of single characters, i.e., R or M (to represent Ringgit Malaysia), a sequence of single digits which may include comma, or decimal, and slash (to represent end of digit amount). The characters after the slash are omitted as they usually do not indicate any significant value of the courtesy amount.

Pre-processing Phase

The objective of the pre-processing phase is to clean the original image from noise or blur and convert it to binary pixels where the pixel '1' represents the object and the pixel '0' represents the background.

We use Matrox Image language library functions (MIL User Guide 1998) for the following process:

- Filtering/smoothing

  Filtering is used to remove noise such as random noise, which may be caused by the camera, the digitiser or uneven lighting during scanning. The MIL Function *MimConvolve*() with parameters M_SPARPEN2 and M_SMOOTH are used to implement this process.

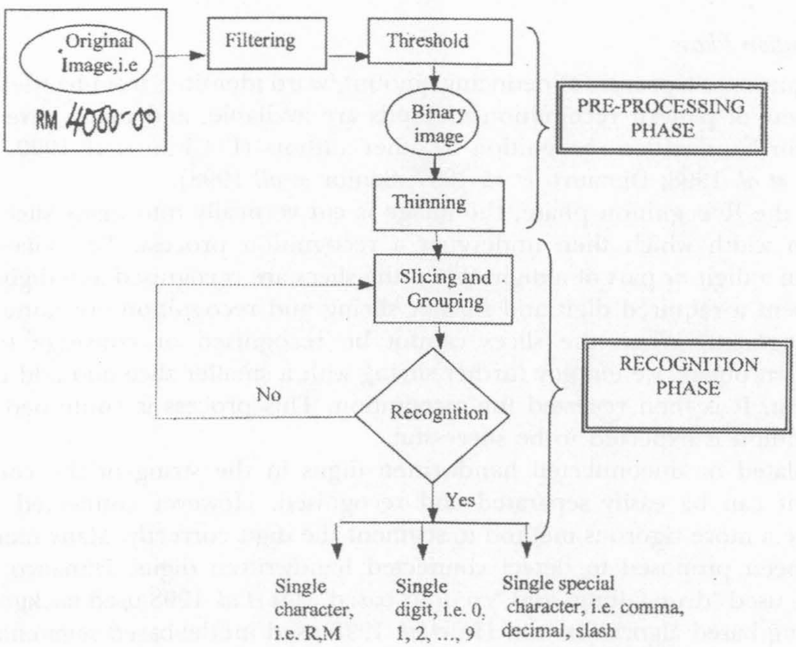


Fig. 2: Process flow to recognise cursive handwritten courtesy amount

- Threshold/binarisation
  To threshold an image means to reduce each pixel to a certain range of values. The MIL Function *MimBinarize*() is used to perform a binarisation operation which reduces a pixel into two scale values (for example, 0 and the maximum value for an 8-bit buffer, 255). A pixel with a 0 value will be seen as black whereas those set to the maximum buffer value will be seen as white.
- Thinning
  The object thickness needs to be reduced to a skeleton size. The purpose of thinning is to simplify the pattern so that analysis such as contour analysis and feature analysis can easily be done. The MIL function *MimThin*() with parameters M_THIN and M_BINARY is used to implement this process.

Fig. 3 shows an example of the pre-processing phase using several MIL functions.

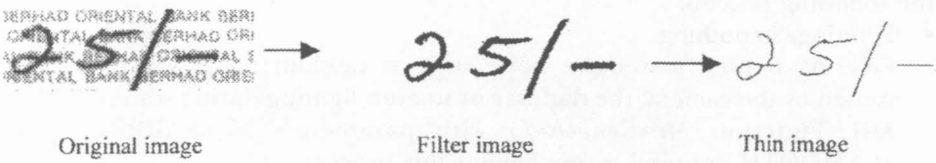


Fig. 3: Example of filtering, binarising and thinning on an image of the figure "25"

Recognition Phase

Recognition is a process of deducing amount/word identities from handwriting. A variety of pattern recognition methods are available, and many have been used for handwritten recognition by other authors (LeChun *et al.* 1990, 1995; Knerr *et al.* 1998; Dimauro *et al.* 1997; Senior *et al.* 1998).

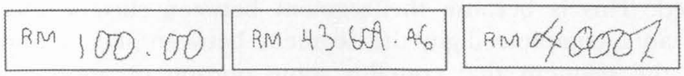
In the Recognition phase, the image is cut vertically into many slices at a certain width which then undergoes a recognition process. The slices may contain a digit or part of a digit. When the slices are recognised as a digit, they represent a required digit and further slicing and recognition are done on a new segment. When the slices cannot be recognised or converge to the unknown object, we employ further slicing with a smaller slice and add to the segment. It is then re-tested for recognition. This process is continued until recognition is expected to be successful.

Isolated or unconnected handwritten digits in the string of the courtesy amount can be easily separated and recognised. However connected digits require a more rigorous method to segment the digit correctly. Many methods have been proposed to detect connected handwritten digits. Dimauro *et al.* (1997) used "drop falling" and "contour-based", Lu *et al.* 1998 used background thinning based algorithm, and Hu *et al.* 1998 used model-based segmentation method for handwritten numeral strings.

To recognise an image, three type of images are considered:

a. When an image contains joint digits as shown in *Fig. 4a.*

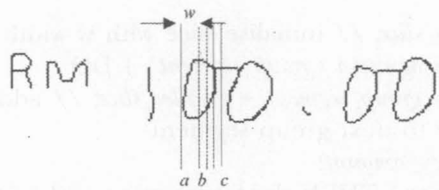
b. When an image contains touched digits as shown in *Fig. 4b.*
c. When an image contains what is required to be slant correction as shown in *Fig. 4c.*



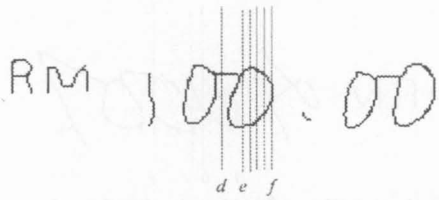
(a) joint digits (b) touched digits (c) image to be slant correction

Fig. 4: Some examples of images to be recognized

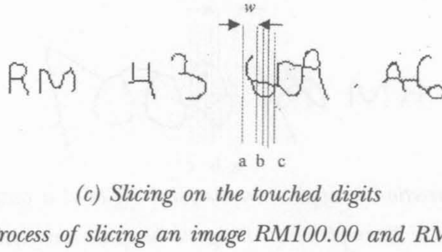
Isolated or unconnected digits, characters R and M, a decimal and a slash are not involved in the process of slicing. They can be separated using the blob analysis technique (Sulaiman *et al.* 2001). The process of slicing of *Fig. 4a* is shown in *Fig. 5a.* The process of slicing started from the slice a to the slice b at a certain width, w. The segment ab is then tested for recognition using a recogniser. The recogniser will inform if it is an unknown object and therefore further slicing but with the smaller slices, i.e., from slice *b* to slice *c*, are added to the segment in order to recognise a digit 0. A similar process of slicing is also applied on the second joint digit which started at the slice d and ended at the slice *f* (*Fig. 5b.*).



(a) Slicing begins at the first of the joint digit



(b) Slicing on the next joint digit



(c) Slicing on the touched digits

Fig. 5: Process of slicing an image RM100.00 and RM43609.46

Fig. 5c shows a similar process of slicing and recognising of the image of *Fig. 4b* which contains the touched digits.

However, when the same process of slicing and recognising is employed on the connected digits of the image of *Fig. 4c,* as shown in *Fig. 6a,* no digit can be identified. This is because the segment between slice *a* and slice *b* is considered an incomplete digit; the segment between slice *a* and slice *c* is considered the segment that contains some portion of another digit. The recogniser will inform that both segments be converge on to the unknown object. Slant correction can then be applied on the connected digits using the following formula:

$$x' = x - (Row - y)\sin\theta \tag{1}$$
$$y' = y \tag{2}$$

where *Row* is the height of the image. The object is sheared horizontally from right to the left as shown in *Fig. 6b* and the similar process of slicing and recognising can be applied on this new image.

Algorithm to recognise on the connected digits is as follows:

Algorithm 1:

READ image

$i = 1$; // start grouping the slices

REPEAT

group_segmenti = w_slice; // initialise slice with w width

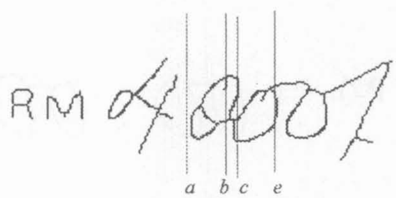
WHILE (NOT *Recognition* (*group_segment*$_i$)) DO

group_segment$_i$ = *group_segment*$_i$ + *smaller_slice*; // add smaller slice

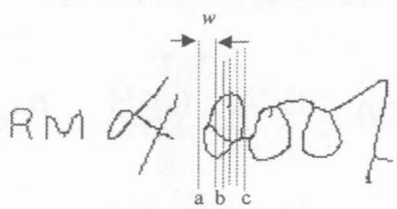
$i = i + 1$; // move to next group segment

UNTIL (*end_of_courtesy_amount*)

IF (*unrecognised the image*) THEN slant correction and GO TO REPEAT.



(a) Slice b and slice c contain some portion of two digits



(b) Slant correction is applied on the joint digits at a certain angle

Fig. 6: Example of an image to be slant correction

FEATURE EXTRACTION

Feature extraction is a methodology of extracting features from the images. In other words, the image is represented by a set of numerical features to remove redundancy from the data and reduce its dimension. These features can be categorized as good features if fulfilling the characteristics below:

- Small interclass invariance whereby slightly different shapes with similar general characteristics should have numerically close values.
- Large interclass separation in which features from different classes should be quite different numerically.

There are many techniques for feature extraction (Mehdi *et al.* 1997; Samer *et al.* 1998). These can be classified as local features or global features. Global features depend on the entire shape of an image for the determination of the feature, while local features describe the limited regions of the shape and are affected by other region of the image. Techniques widely used, as global features are moment invariants, *fourier* descriptors, *eigenvector*, structural analysis and heuristic.

The following methods are used to extract the most discriminant features from the image :

- Number of transitions.
- Shapes.
- Moment invariant.

The number of transition and shape are based on local feature extraction whereas moment invariant is based on global features. The number of transition means that the numbers of pixel '1' for each row and column of the image are counted. The image is divided into 5 partitions on row and 5 partitions on column which bring a total of 10 features. For each partition, an average count of pixel '1' is calculated.

To measure the shapes, the image is divided into 5 partitions on its row and 5 partitions on its column. The image is then viewed from the top, horizontal and bottom directions. An average length from pixel '0' until pixel '1' on each view is calculated. About 20 features are extracted where 5 are on each sides.

Moment invariant method determines the global feature of the image. This method was presented by Hu (1962) in his historical paper on the use of moment invariant in 2-D pattern recognition. Shamsuddin *et al.* (2000) has improved the scaled-invariant moment formulation and successfully applied it on unconstrained isolated handwritten digits. There are 9 moment values calculated up to the $4^{th.}$ order which have been extracted from the image.

A total of 39 features (10 features from the number of transitions, 20 features from the shapes, and 9 features from the moment invariants) are extracted. These features are normalised between -1 and 1. The output is a binary number ranging from 0 to 9 representing digit 0 to digit 9.

Table 1 shows the global feature extraction values (9 features) using the moment invariant method, Table 2 shows the local feature extraction values (10 features) based on the number of transitions and Table 3 shows local feature extraction values (20 features) based on the shapes.

TABLE 1
Global feature extractions using moment invariant on the selected image digits

Digits	Moment values								
	η_{02}	η_{03}	η_{11}	η_{12}	η_{13}	η_{21}	η_{22}	η_{30}	η_{31}
0	1.0000	-0.3615	-0.8292	-0.8069	-0.8094	-1.0000	-0.2835	-0.7833	-0.8502
1	0.7582	0.1215	0.9600	0.0184	0.5850	-0.3074	0.6281	-1.0000	1.0000
2	1.0000	-0.4574	-1.0000	-0.1249	-0.9898	-0.4403	0.5157	-0.6984	-0.7552
3	1.0000	-0.6463	0.3114	-0.6584	0.3631	-1.0000	0.3227	-0.4968	-0.0771
4	1.0000	-0.7774	-0.4314	-1.0000	-0.4724	-0.8877	-0.3102	-0.9008	-0.4790
5	1.0000	-0.6340	-0.6550	-1.0000	-0.9176	-0.0473	0.2464	-0.5717	-0.9893

TABLE 2
Local feature extractions based on the transitions of the selected image digits

	Feature extraction method based on the transitions				
Digit 0					
View from the top	-0.333333	-0.111111	-0.111111	-0.111111	-0.333333
View from the bottom	-0.259259	-0.111111	-0.111111	-0.111111	-0.222222
Digit 1					
View from the top	-0.777778	-1.000000	-1.000000	-1.000000	-1.000000
View from the bottom	-0.777778	-0.777778	-0.777778	-0.777778	-0.777778
Digit 2					
View from the top	-0.857143	-0.785714	-0.714286	-0.642857	-0.750000
View from the bottom	-0.809524	-0.857143	-0.857143	-0.857143	-0.857143
Digit 3					
View from the top	-0.878788	-0.81818	-0.575758	-0.696970	-0.797980
View from the bottom	-0.797980	-0.878788	-0.878788	-0.878788	-0.878788
Digit 4					
View from the top	-0.925192	-0.980605	-0.934428	-0.925192	-0.980605
View from the bottom	-0.954746	-0.915957	-0.915957	-0.961211	-0.980605
Digit 5					
View from the top	-0.789474	-0.649123	-0.578947	-0.578947	-0.824561
View from the bottom	-0.859649	-0.859649	-0.859649	-0.672515	-0.859649

NEURAL NETWORK MODEL

We use the popular multilayer Neural Network (NN) trained by the backpropagation (BP) algorithm (Rumelhart *et al.* 1987; LeChun *et al.* 1990) as the recogniser with the new error function (Shamsuddin *et al.* 2001). BP model is basically a gradient descent method and its objective is to minimise the mean square error between the target values and the actual outputs. Thus the mean squared error function (MSE) is defined as:

TABLE 3
Local feature extractions based on the shapes of the selected image digits

Feature extraction method based on the shapes					
Digit 0					
View from the top	0.111111	-1.000000	-0.777778	-0.33333	1.000000
View from the left side	-0.407407	-0.851852	-1.000000	-1.000000	-0.555556
View from the bottom	-0.111111	-1.000000	-1.000000	-0.333333	0.555556
View from the right side	0.481481	-0.55555	-1.000000	-1.000000	0.111111
Digit 1					
View from the top	1.000000	-1.000000	-1.00000	-1.000000	-1.000000
View from the left side	-1.000000	-1.000000	-1.00000	-0.925926	-0.733333
View from the bottom	-0.481481	-1.000000	-1.000000	-1.000000	-1.000000
View from the right side	-0.555556	-0.555556	-0.555556	-0.629630	-0.822222
Digit 2					
View from the top	1.000000	0.071429	-1.000000	-1.000000	-0.821429
View from the left side	-0.190476	0.523810	0.42857	0.142857	-0.500000
View from the bottom	-0.928571	-1.000000	-1.000000	-1.000000	-0.035714
View from the right side	-0.809524	-0.952381	-0.904762	-0.571429	-0.392857
Digit 3					
View from the top	-0.818182	-1.000000	-1.000000	-0.454545	0.292929
View from the left side	-0.878788	-0.393939	-0.515152	-0.151515	-0.099567
View from the bottom	1.000000	0.090909	-1.000000	-0.818182	-0.595960
View from the right side	-0.515152	-0.434343	-0.272727	-0.79798	-0.774892
Digit 4					
View from the top	-0.075522	0.940389	0.968095	0.044541	1.000000
View from the left side	-0.670291	-0.786659	-1.000000	0.002057	0.073668
View from the bottom	0.377020	0.330843	-0.269468	0.303136	0.271231
View from the right side	0.480458	-0.211284	-0.023803	-0.030267	0.292479

$$E = \frac{1}{2}\sum\left(t_{kj} - o_{kj}\right)^2 \tag{3}$$

where t_{kj} is the target output from node *k* to node *j*,
o_{kj} is the actual output from node *k* to node *j*.

The new error function (E_{new}) is used to increase the convergence rates of BP learning and defined implicitly as (Shamsuddin *et al.* 2001):

$$E_{new} = \sum_{p,j} g_{p,j} \tag{4}$$

with $g_{p,j} = \dfrac{e_{p,j}^2}{1 - a_{p,j}^2}$,

where $e_{p,j} = t_{p,j} - a_{p,j}$,
and $e_{p,j}$ is an error at output unit j for pattern p,

$t_{p,j}$ is the target value of output unit j for pattern p,
$a_{p,j}$ is an activation of output unit j for pattern p.

The derivatives of the improved error function which are used to determine a descent direction are defined in the same way as for the mean square error function.

EXPERIMENTS AND RESULTS

We have experimented on several hundred digits from the selected cursive handwritten courtesy amounts of Malaysian bank cheques. The digits may be unconnected, joint or touched between two or more digits. Appendix A shows some examples of the original images.

We divided the experiments into two parts :

a) Recognition.
b) Slicing and recognition.

Recognition

Firstly, the NN are trained with more than 300 isolated handwritten digits from various samples of selected Malaysian bank cheques. Some examples of the isolated digits served as the training data set are shown in *Fig.* 7. More examples of the training data set are shown in Appendix B.

A three layer NN architecture with the new error function of BP learning algorithm (Shamsuddin *et al.* 2001) is used. The input data for the BP model

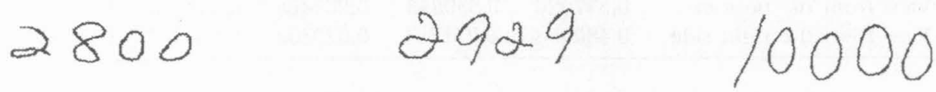


Fig. 7: Some examples of training data set

are the numerical features extracted from the isolated digits. Decimal, comma, and slash are not extracted because they can be detected easily during the programming process.

The input nodes for the network architecture are 39, the hidden layer contains 39 nodes, and the output layer contains 12 nodes. The output nodes are used to represent a binary value for digit 0 to 9, the unknown objects such as the incomplete digits, and the rejected objects such as the objects after slash. All initial weights and biases are set randomly between -1 and 1. For training purposes, the learning and momentum rates are set to 0.01 and 0.9 respectively with a sigmoid activation function. To terminate the training cycles, the minimum sum of errors is set to 0.01.

Further experiment is also carried out to compare the use of the new error function of BP with the standard error of BP and Kalman's BP (Kalman *et al.* 1991) on the recognition of the isolated digits. The same learning and momentum rates are used for this experiment. The result from the experiment is shown in *Fig. 8*. It can be seen that the new error function of BP gives better results with faster convergence rates compared to the standard BP and Kalman's BP. The new learning rate takes 345 cycles to converge to the solution, compared to the standard BP, which takes 845 cycles, and Kalman's, which takes 1553 cycles.

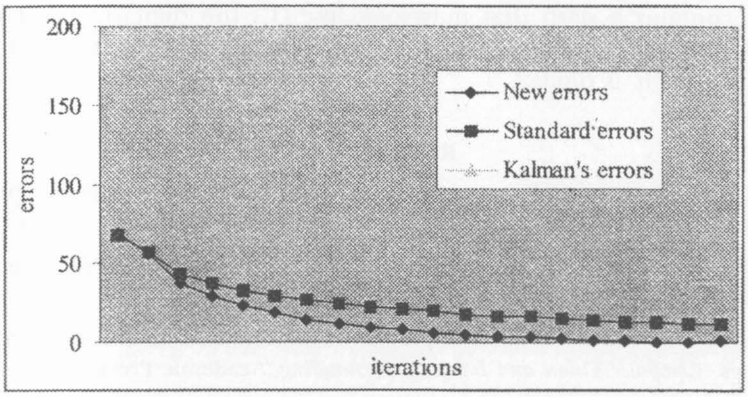


Fig. 8: Comparison of the convergence rates

Slicing and Recognition

About 156 digits from the original courtesy amounts of Malaysian bank cheques are used for testing. These figures are fed into the trained neural network. A 95% recognition rate has been achieved on tested data with 4% classified into the unknown objects and 1% classified as incorrect digits.

Although our method has been successfully tested on quite a number of cursive handwritten courtesy amounts there are still cases where recognition has not been successful. The reasons for not being successfully recognised are:

- The amount contains slash that may look like '1'.
- The digit joint with the decimal.
- Incompletely written digit 0 may be classified into the wrong digit such as digit 2.

There is no direct solution to overcome the above difficulties. However, the percentage of error for the unsuccessful recognition is small which is 5%.

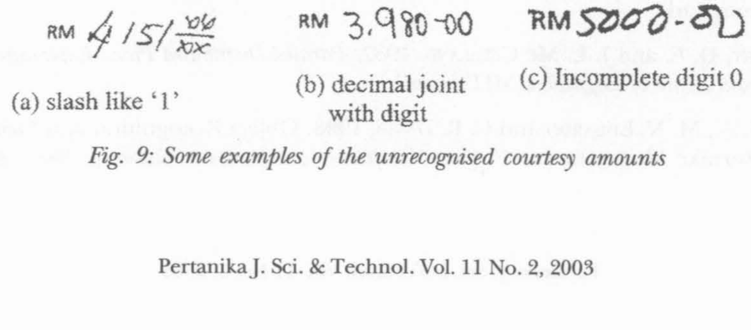


Fig. 9: Some examples of the unrecognised courtesy amounts

CONCLUSION

We have described a technique called grouping single slice segment mechanism to segment and recognise cursive handwritten courtesy amounts of Malaysian bank cheques. We have implemented and tested on more than hundred courtesy amounts of selected Malaysian bank cheques. A 95% recognition rate has been achieved with 4% converge to the unknown images, and 1% converge to the incorrect digits. The reasons for not being successfully recognised are the amount contains a slash that may look like '1', the digit is joined with the decimal, and the incompletely written digit 0 which may be classified into the wrong digit such as digit 2.

REFERENCES

DIMAURO, G., S. IMPEDOVO, G. PIRLO and A. SALZO. 1997. Automatic bankcheque processing: A new engineering system. *Automatic Bank Cheque Processing,* World Scientific, p. 5-42.

HU, M.K. 1962. Visual pattern recognition by moment invariant. *IRE Trans. Inform. Theory,* IT-8: 179-187.

HU, J. and H. YAN. 1998. Model-based segmentation method for handwritten numeral strings. *Computer Vision and Image Understanding.* Academic Press.

KALMAN, B. L. and S. K. KWASNY. 1991. A superior error function for training neural network. *Proceeding of International Joint Conference of Neural Networks* **2**: 45-52.

KNERR, S., E. AUGUSTIN, O. BARET and D. PRICE. 1998. HMM based word recognition and its app. to legal amount reading on French checks. *Computer Vision and Image Understanding* **70(3)**: 404-419.

LECUN, Y., B. BOSER, J. S. DENKER, D. HENDERSON, R. E. HOWARD, W. HUBBARD and D. L. JACKEL. 1990. Handwritten Digit Recognition with a Back-Propagation Network. In *Advances in Neural Information Processing Systems* **2**, ed. D. Touretzky, Morgan Kaufman.

LECUN, Y., L. JACKEL, L. BOTTOU, A. BRUNOT, C. CORTES, J. DENKER, H. DRUCKER, I. GUYON, U. MÛLLER, E. SÄCKINGER, P. SIMARD and V. VAPNIK. 1995. Comparison of Learning Algorithms for Handwritten Digit Recognition. In *International Conference on Artificial Neural Networks,* ed. F. Fogelman and P. Gallinari, p. 53-60, Paris.

LU, Z. and Z. CHI. 1998. Background-thinning based algorithm for separating connected handwritten digit strings. In *Proceeding IEEE Intern. Conf. on Acoustics, Speech and Signal Processing.* p. 1065-1068. USA.

MEHDI, D. and F. KARIM. 1997. Farsi Handwritten Character Recognition with Moment Invariants. In *International Conference of Digital Signal Processing,* p. 507 - 510.

MIL USER GUIDE. 1998. *Matrox Imaging Library User Guide* (version 5.1). Matrox Electronic Systems Ltd.

RUMELHART, D. E. and J. L. MC CLELLAND. 1987. *Parallel Distributed Proc.: Explorations in the Microstructure of Cognition.* MIT Press.

SAMER. M. A., M. N. EDUARDO and C. R. DAVID. 1998. Object Recognition and Orientation via Zernike Moments. In 3rd *Asean Conference on Computer Vision,* p. 386 - 393.

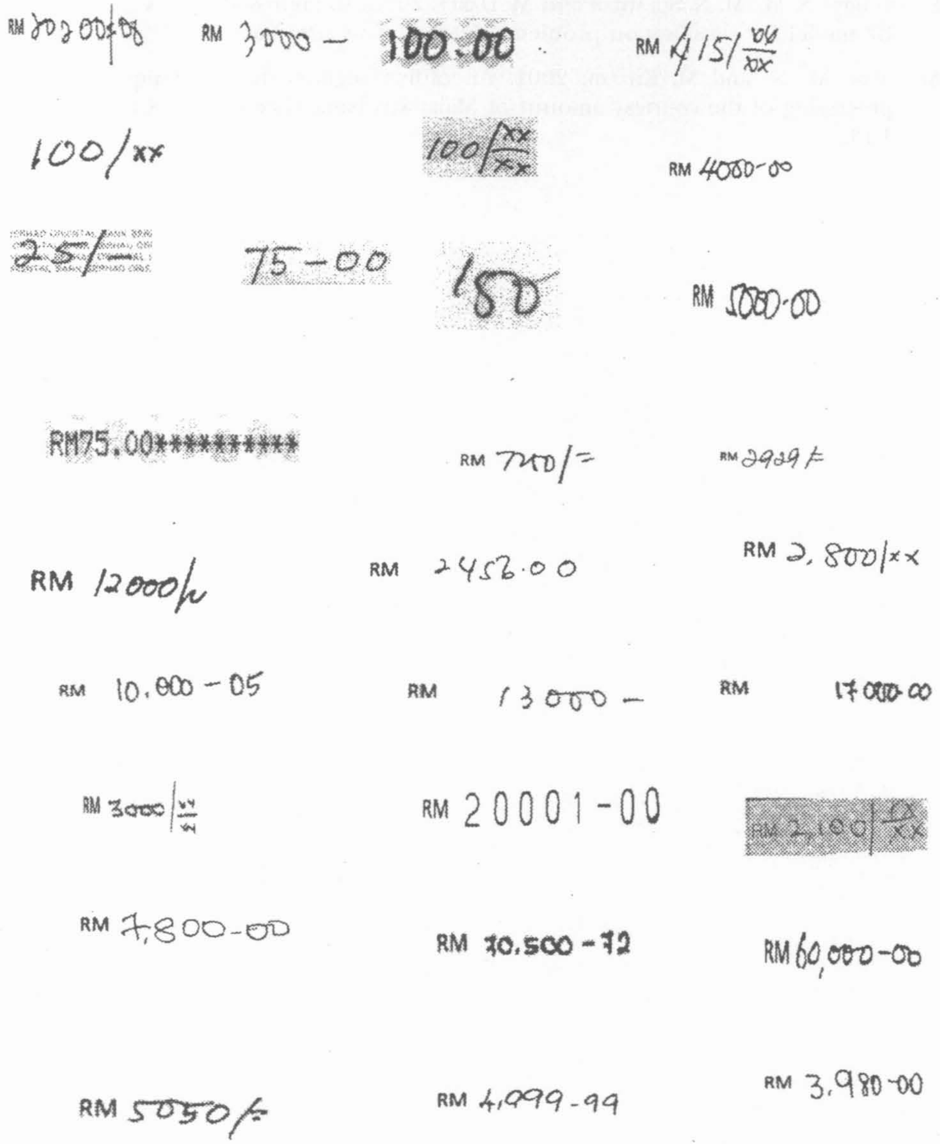
SENIOR, A. W. and A. J. ROBINSON. 1998. An off-line cursive hand writing recognition system. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20(3)**.

SHAMSUDDIN, M., M. N. SULAIMAN and M. DARUS. 2000. Improved scale-invariant moments for deformation digits. *Intern. J. Computer Math.* **74**: 439-447.

SHAMSUDDIN, S. M., M. N. SULAIMAN and M. DARUS, 2001. An improved error signal for the BP model for classification problems. *Intern. J. Computer Math.* **76**: 297-305.

SULAIMAN, M. N. and M. KHALID. 2001. An offline segmentation technique for the processing of the courtesy amount of Malaysian bank cheques. *ELEKTRIKA* **4(2)**: 1-13.

APPENDIX A:
Some examples of the courtesy amounts taken from Malaysian bank cheques



APPENDIX B:
Some examples of images for training data set