An Intelligent System for Conflict Resolution in Handwritten Address Recognition

Chin Keong Lee and Graham Leedham

School of Computer Engineering
N4-2A-32 Nanyang Avenue
Nanyang Technological University
Singapore 639798
asgleedham@ntu.edu.sg

Abstract

Background: The basic recognition engine of a handwritten address interpretation system, for use in postal sorting automation, is an OCR algorithm that recognises a numeric or alphanumeric string, such as a postcode, and matches it against a set of valid postal delivery points. However, the OCR system is highly vulnerable to errors due to the uncertainty that arises when the imperfect OCR result of the alphanumerics are combined. These errors result in wrongly sorted mail which costs postal organizations throughout the world a significant amount of money to resolve after the error is detected by the postal delivery man and also delays the delivery of the wrongly-sorted item. A generic expert system model that forms part of a handwritten address interpretation system for conflict resolution in unconstrained handwritten address recognition is presented in this paper, to reduce this error.
Method: The proposed expert system resolves the conflicts and reduces the error rates by fusing a holistic pattern recognition method with expert knowledge based on *a posterior* information. The system was evaluated using 1,071 handwritten Singapore addresses.
Results: Experimental results show that the expert system achieved a significant reduction in error rates. Performance was improved from 71.2% correctly sorted, 4.8% reject (cannot sort) and 24.0% error (wrongly sorted) rates using OCR only to 63.7% correctly sorted, 35.7% reject (cannot sort) and 0.6% error (wrongly sorted) rates using the proposed approach.
Conclusions: The error rate (proportion of wrongly sorted mail) can be significantly reduced using this method and significantly alleviate the need to carry out expensive resorting.

Keywords: Handwritten address recognition; OCR error recovery

I. Introduction

Automatic sorting of handwritten mail pieces is a very challenging task. The main problems in handwritten address interpretation are parsing and recognising a set of correlated entities, such as the postcodes, street names and building numbers, in the presence of incomplete information. It is a computer vision problem which has stringent performance requirements in commercial applications. The task of accurately recognizing and interpreting a handwritten address is complicated by the variability and complexity of the addresses, word shape distortion due to non-linear shifting, unpredictable writing styles and failure to locate the actual address in the address database due to severe postcode recognition errors and intrinsic deficiencies in the address database. The use of the postcodes in handwritten addresses was described by Srihari et al. [15], and works well in addresses where the postcode is correctly located and recognised. A more robust approach to postcode recognition and error detection is to correlate the *redundant* information found in the remainder of the handwritten address with the postcode. For example, Madhvanath et al. [12] and Srihari & Kuebert [16] used the city and state information, and Gader et al. [5] used the street number information to verify that a postcode had been correctly recognised. Others have described the use of a syntactic (structural) verification approach. For example, Downton & Kabir [3] and Tregidgo & Downton [17] used structural features extracted from the appropriate parts of the address image and compared these against expected features corresponding to the postulated postcode(s). If the uncertainty or confusion in the candidate postcode could be resolved, most of the OCR and substitution errors could be corrected or avoided.

In this paper we present a generic expert system model for conflict resolution in unconstrained handwritten address recognition. We use Singapore postal addresses in this work but the technique is extensible to other country's address formats and other constrained vocabulary text recognition applications where redundancy can be exploited to verify and correct errors. A neural network OCR classifier returns a confidence value for each of the six postcode digits and outputs candidate postcodes. Based on the information returned by the neural network classifier and a dictionary search of existing postcodes and addresses the inference engine of the expert system decides whether there is confusion or not. If there is no confusion the expert system accepts the candidate postcode and outputs it as the recognised postcode without invoking the computationally expensive expert system. In case of confusion, the expert system is used to resolve the conflict between the candidates, by querying the delivery point files that contain all possible valid addresses to find the best interpretation based on the postcode and features extracted from the rest of the handwritten address.

II. Feature Extraction

There are many variations in the handwritten versions of the same address. In order to recognise the address, features that are stable and invariant to this natural variation need to be selected. Invariant features should have approximately the same values for all samples of the same class (word). However, not all variations of handwritten addresses from the same class exhibit invariant features. For example, some writers have very distinct ascenders and descenders in such letters as ''t', 'b', 'y', 'g' while others write them very short making them difficult to distinguish from other letters. If invariant features cannot be found, an alternative is to pre-process the input image to improve the image quality by carrying out such operations as skew normalisation, slant correction and thinning. However, this introduces new discretization errors. The phenomenon called the *curse of dimensionality* [9] cautions us that in the statistical classification approach, the number of features must be kept reasonably small if a limited training set is used.
In this work we used the following features:

global features -	line and word positions and the number of words in the address,
local features -	number of characters, loops, ascenders, descenders and ascender/descender sequence in each word.

This set of features was used because they are invariant and highly relevant for address similarity comparison in the sense that they minimise the within-class pattern variability while enhancing the between-class pattern variability. The line and word positions and number of words were extracted during the line and word segmentation processes. Since some of the features used are different for upper case and lower/mixed case words (e.g. ascenders and descenders are found only in lower/mixed case words), identification of the word case was first applied before feature extraction took place. The technique for word case classification has been independently described elsewhere [8]. Main body determination of a word is essential in this work and was used in extracting ascenders and descenders, determining word case and in character segmentation. The standard method to determine the main body of the word is based on its horizontal histogram [14]. This method is relatively accurate in determining the main body and ascenders/descenders for handwriting written carefully in a straight line. However, methods based on horizontal histograms will fail for skewed words. Therefore baseline detection and correction for skewed words is necessary before implementing this method.

A. *Estimation of the number of characters per word*

In upper case words, the area, height and aspect ratio of each component were checked against corresponding threshold values to determine a single or multiple character word. The number of letters in a component with multiple characters was calculated as the rounded value of the width of this component divided by its aspect ratio. In lower case words, a simple method was used based on counting the number of line crossings through the main body of the word. The number of line crossings is the number of black runs encountered by scanning each row in the main body. The estimate of the number of letters in the word was calculated as the rounded average value of the number of line crossings.

B. *Ascenders and descenders detection*

In a mixed case word, the conventional method of detecting the ascender and descender information is by first identifying the ascender zone, descender zone and main body zone of the word. Connected component analysis is then applied to the ascender and descender zones. A heuristic test based on component attributes then attempts to determine whether ascenders or descenders exist. If they exist, the number of ascenders and descenders are recorded. From their relative positions within the word, information about the ascender and descender sequence can be extracted. Because some words are slanted, slant correction must be applied to improve the accuracy in extracting these features [1], [10]. The effect of the algorithm is shown in Figure 1.

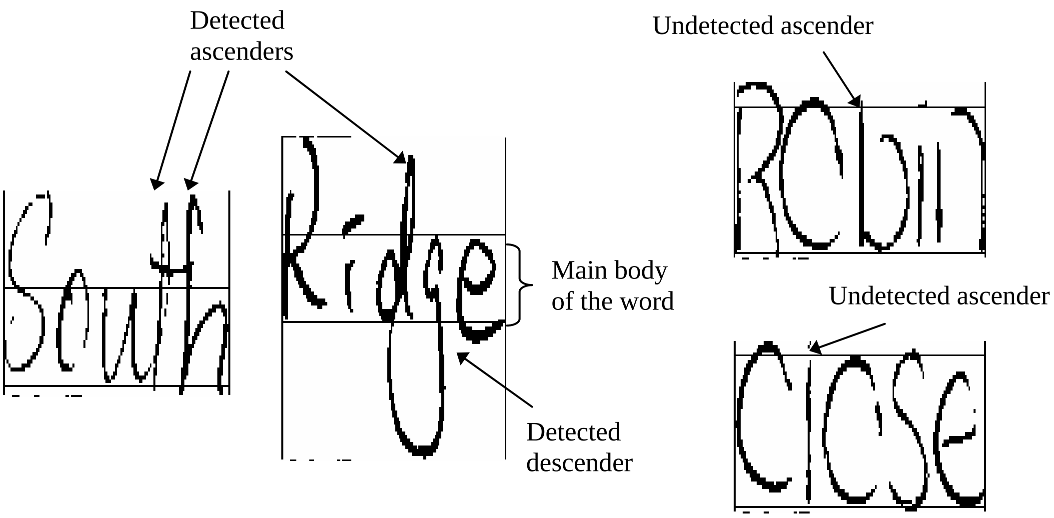

Figure 1: Detection of ascenders and descenders and examples of writing where detection is not possible.

C. *Closed loop and near loop detection*

Loops were detected by utilising two independent filtering steps: top-down filtering and bottom-up filtering [2]. Top-down filtering finds face down valleys in the background region and bottom-up filtering seeks face up valleys. By combining a top-down and bottom-up filter, loop regions can be identified. In handwriting, loops are frequently broken or not fully closed. In these cases another algorithm is employed to determine broken loops as being centre cavities based on mathematical morphology [6], [13]. The underlying theory of mathematical morphology is that of set theory and performs the two operations, *dilation* and *erosion*, on digital images based on shape. If used appropriately, mathematical morphology operations tend to simplify image data preserving their essential shape characteristics and eliminating irrelevancy [7]. The overall effect after applying these algorithms is shown in Figure 2.

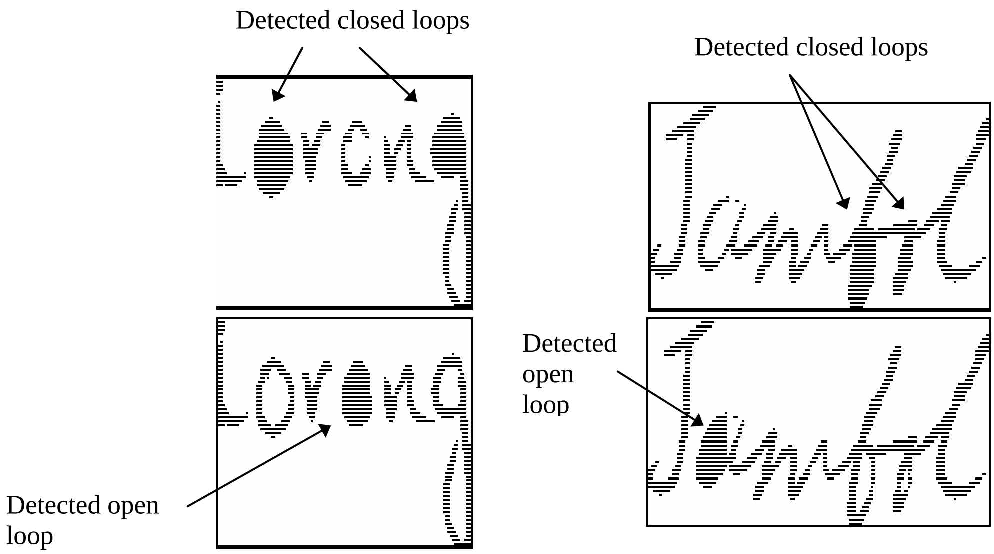

Figure 2: Detection of open and closed loops

D. *Word similarity measure*

The distances for the features such as line and word position, number of ascenders, number of descenders, number of characters and number of loops are the differences between values extracted from the address image and ideal values for particular words derived from a postulated address. The distance for an ascender/descender sequence, $D(u,v)$ is derived based on the similarity measure between two strings:

$$D(u,v) = \left[\sum_{(\alpha)} \left| p(u:\alpha) - p(v:\alpha) \right| \cdot |\alpha| \right]$$

where u and v are two ascender/descender sequences of m and n ordered elements, respectively, where the element can be either '1' (ascender) or '0' (descender), $p(s:\alpha)$ denotes the number of times sub-ascender/descender sequence α occurs in ascender/descender sequence s, including partial overlaps, and $|\alpha|$ is the length of sub-ascender/descender sequence α [4].

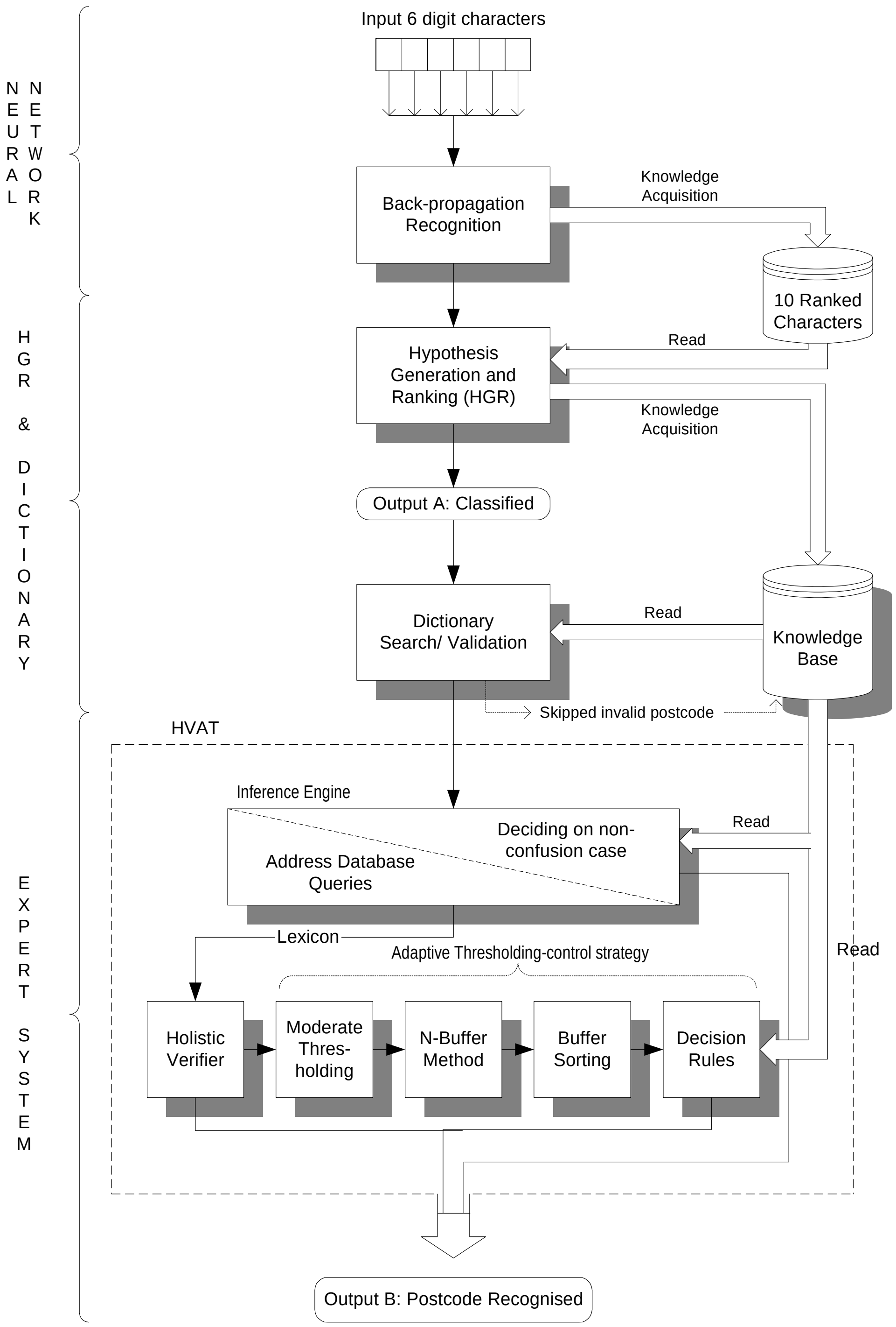

Figure 3: The architecture of the address interpretation expert system.

III. Proposed Design

A. *Expert System*

An important consideration during the design was that a decision of ACCEPT or REJECT be reached as early as possible during the processing of a mail-piece. This observation suggests a need for a hierarchical strategy when calling the more computationally expensive modules. For example, the expert system is called if, and only if, an ACCEPT/REJECT decision cannot be made based solely on the results of OCR or the dictionary validation.

The proposed architecture of the address interpretation expert system is shown in Figure 3. The expert system is the verification module HVAT. This purely software Holistic/Analytic Verifier Adaptive Thresholding-Control module that is cascaded to the output of the neural network classifier has expert-like ability to solve a domain specific problem.

In our case, the main function of the expert system is to resolve the confusions or uncertainties that occur in the class classification from the neural network. A cluster is formed based on the highest class rank of the neural network results at Output A, which comprises six classes that are associated with the six-digit postcode. Confusion can arise when this cluster is checked against a postal address database and found to be invalid or contains address dissimilarities. If there is no confusion then the cluster is assigned as the recognised postcode at Output B with a higher confidence level to strengthen the result given by the neural network. If there is confusion, then the particular clusters among which the confusion occurs should be identified (other clusters are formed by the HGR unit as shown in Figure 4). The conflict between the clusters is then resolved by invoking the Adaptive Thresholding-Control modules to make one of three decisions- ACCEPT, REJECT or RECOVERED.

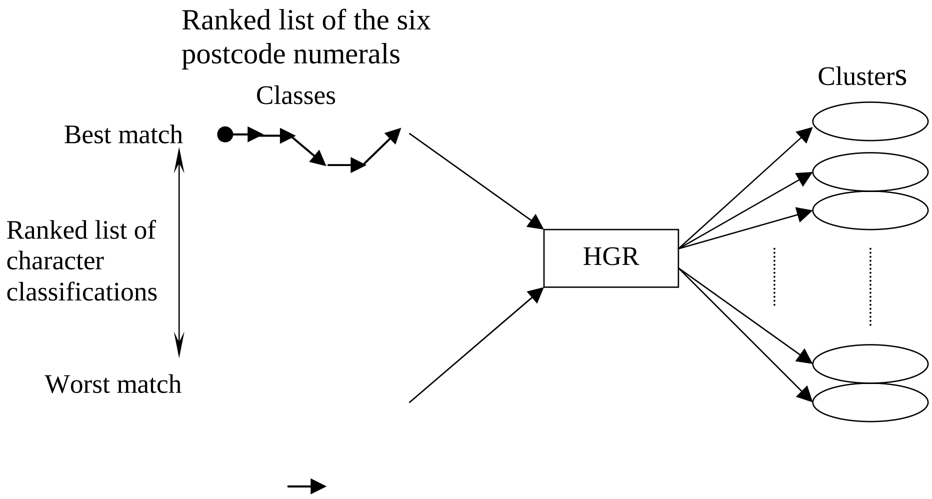

Figure 4: Generation of postcode clusters

Five types of confusion may occur in the system:

*C*1: at Output A when the cluster confidence is less than a predefined threshold value;
*C*2: the cluster is invalid, i.e. it does not have a valid entry in the postal dictionary;
*C*3: address dissimilarities, i.e. the address image does not have a high confidence match with the postulated address(es) associated with the cluster;
*C*4: the difference between the maximum and next maximum address confidence is less than a predefined threshold value;
*C*5: the conflict that occurs inside the *N*Buffer where the top cluster does not have a maximum address confidence.

The input to the neural network classifier is an array of six numerals. The confidence values returned from the neural network for each numeral are represented as an array of 10 elements. Each element of the array consists of a digit label and a confidence value and the arrays are stored in a database of 10 ranked characters. The hypothesis generation and ranking unit attempts to hypothesise all the possible combinations of postcodes and this information is stored in a knowledge base. The knowledge base is represented as a record and consists of the ranked hypothesis, its confidence and a pointer to a single linked list. The postulated addresses that belong to a given cluster are added to the linked list of the corresponding cluster. The features vector of the postulated address is obtained by querying the address database.

When the confusion is of types *C*1 or *C*2, the address confidence is computed in the Holistic/Analytic Verifier module for the top *N* clusters. An important aspect of the system is that the lower rank clusters tagged with address confidence, are buffered and examined for conflict resolution as shown in Figure 5.

Thus, the *a posterior* knowledge of the next *N* –1 top-ranked clusters is formed inside an *N*-Buffer which represents the expert knowledge that will be used to produce a correct interpretation of the postcode at the decision rules are applied. In addition, if the top-ranked cluster is invalid, this information is added to the subsequent cluster's record. It is capable of correcting a decision which was wrongly classified by the neural network classifier and the holistic verifier by reconfirming the address confidence against an analytical result of the building number or a recovery threshold [17]. These novel characteristics are essential, particularly for reducing classification errors and decorrelating information.

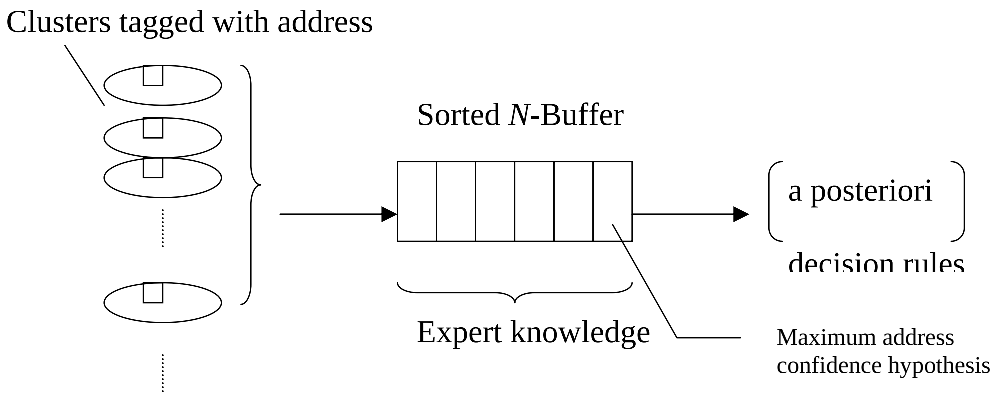

Figure 5: Buffering of clusters

In cases when the expert system cannot make a firm decision, that is if confusions of type *C*3, *C*4 and *C*5 exist, the mail piece is rejected and would have to be hand-sorted. If the address confidence of the top cluster exceeds a minimum threshold, then the postcode is assigned to the same cluster as that assigned by the neural network with a higher confidence value. Otherwise it is recovered if, and only if, the address confidence exceeds a maximum threshold or confusion of type *C*5 does not exist. However, recovery of the errors caused by poor quality images that contain severe broken strokes and ambiguous handwriting is impossible.

B. *N-Buffering Methodology*

The use of thresholding to find all possible postcode hypotheses with the intention of identifying which is the actual postcode at a later time is termed *N*-Buffering. The number of possible hypotheses to be buffered is determined by the buffer size, *N* that is adjustable to achieve an optimum overall performance.

Let P_1 denote the confidence of the correct postcode and T_1 be the address confidence of the correct postcode. Assume that there exists a spurious postcode that is denoted by P_2 with address confidence T_2, and T_{acc} is the *accept* threshold. Below, we examine the case for using the *N*-Buffering method.

Note that T_{acc} is one of the main parameter used to control the performance of the system. It can be set based on the requirement of the postal sorting. An increase in this parameter will results in a higher rejection (failed to automatically sort the address) but at the same time ensures minimum errors (wrongly sorted address). This method of simply reducing the error rate by increasing the rejection rate is not the intention of the research. A reasonably low T_{acc} is chosen throughout the experiments to allow higher level of fairness to the postcode recognition by avoiding strict rejection using the handwritten address features. As such, theoretically, the efficiency of the intelligent system proposed can be evaluated by lowering the T_{acc}.

Let $P_1 > P_2$, $T_1 > T_2$, and let $T_1 = 0.70$, $T_2 = 0.61$. That is, the correct postcode confidence is highest but the address confidence is highest for the spurious address and postcode.

(*A*) *Not using the N-Buffering method* (*N=1*):
Generally, there are three cases to be considered
Case I: If T_{acc} is high, say $T_{acc} = 0.80$,
$\because\ T_1 < T_{acc}$,
$\therefore\ P_1$ is wrongly rejected which results in low recognition and low error rates.
A higher T_{acc} threshold value is able to block out most of the errors, but it has a higher tendency to miss the correct choice of postcode.

Case II: If T_{acc} is low, say $T_{acc} = 0.60$,
$\because\ T_1 > T_{acc}$,
$\therefore\ P_1$ is correctly verified which results in higher recognition and higher error rates.

A lower T_{acc} threshold value results in a looser condition that is more tolerant to noise. This may increase the overall recognition rate but more spurious hypotheses may be wrongly accepted.

Case III: If $P_2 > P_1$ for the same T_1, T_2 and T_{acc} as in Case II,
$\because\ T_2 > T_{acc}$,
$\therefore\ P_2$ is wrongly verified which results in higher recognition and higher error rates.

Due to the classification errors in the neural network, the correct postcode takes a lower confidence. As the spurious postcode has an image in the address database but is different from the actual address, it usually matches with a lower address confidence. The top-choice classification performance based on different *accept* threshold values, T_{acc} on a test set is tabulated in Table 1. As can be seen, the recognition rate deteriorates to below 70% as the *accept* threshold value T_{acc} increases. Thus, it is very difficult to achieve the requirement of low error and high recognition rates.

T_{acc}	Recognition rate	Rejection rate	Error rate
0.60	79.3%	15.9%	4.8%
0.70	74.0%	22.1%	3.9%
0.80	66.3%	31.2%	2.4%

Table 1: Top-choice classification performance for different threshold value, T_{acc}

Using the N-Buffering method.
Let N = 2 and consider the cases

In Case II if T_{acc} is low, say $T_{acc} = 0.60$, and if $T_1 > T_2$ and the confidence of postcode P_1 is greater that the confidence of postcode P_2 ($P_1>P_2$) then
$\because\ T_1 > T_2$,

P_1 is correctly verified

And similarly for Case III: when $P_2 > P_1$ and $T_2<T_1$ then

$\because T_2 > T_1$.

In this case confusion exists and P_2 is correctly rejected.

The result is high recognition and low error rates.

In Case II if T_{acc} is low, say $T_{acc} = 0.60$, and if $T_1 < T_2$ and the confidence of postcode P_1 is greater that the confidence of postcode P_2 ($P_1>P_2$) then

$\because T_1 < T_2$,

then P_1 is wrongly rejected

And similarly for Case III: when $P_2 > P_1$ and $T_1<T_2$

$\because T_2 > T_1$,

then P_2 is wrongly verified.

The above examples justify that the N-buffering method is able to prevent accepting a spurious hypothesis in Case III, while achieving the correct result in Case II. From the experimental results, the problem cases illustrated in B(ii) do not occur when both neural network and expert system fail to assign a higher confidence to the actual postcode. Therefore, this drawback is insignificant.

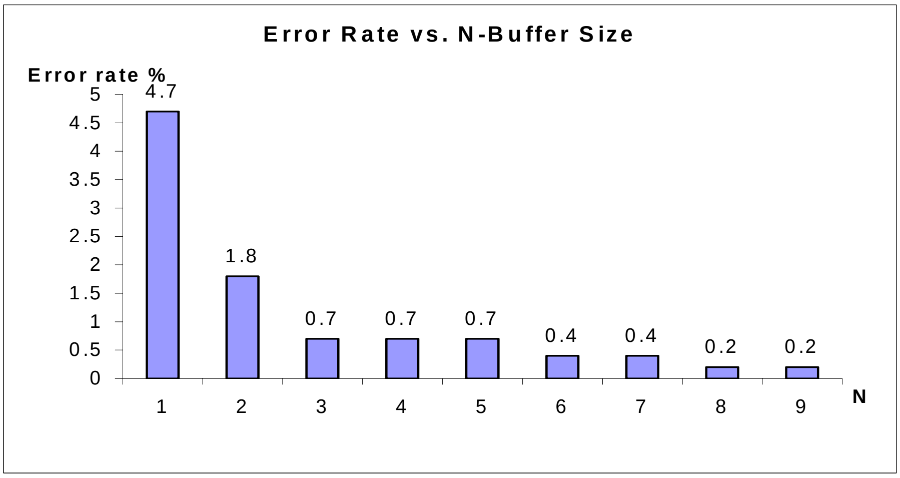

Figure 6(a): Reduction of the error rate as *N* increased on training set

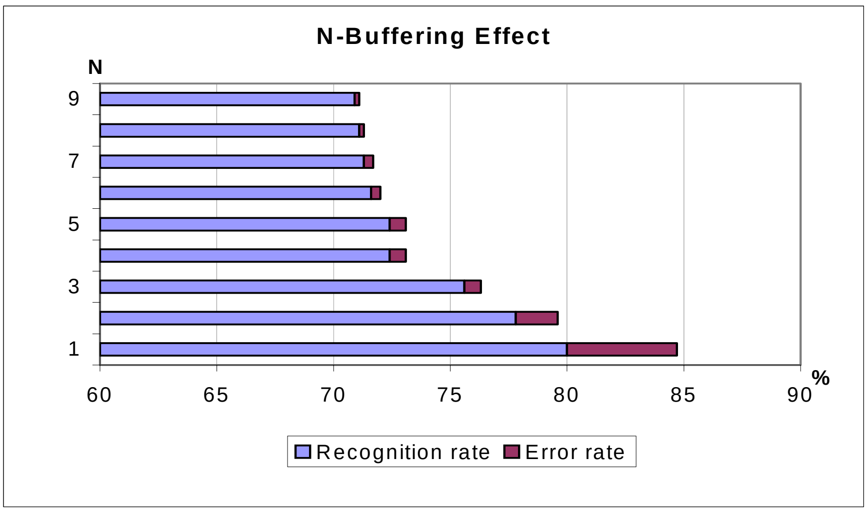

Figure 6(b): Overall performance of *N*-Buffering on the training set

Figure 6(a) shows how the error rate reduces as the buffer size N is increased. The value N was selected as the point at which the error rate dropped to a minimum value. A minimum error rate of 0.2% was obtained for buffer size N=8 onwards using the training set. Figure 6(b) illstrates the change in system performance for different values of N. The value of N was chosen as the saturation point where the error rate achieved the minimum value, that is, at N = 8. As a consequence, performance under rejection, rather than absolute performance, is the yardstick for measuring the effectiveness of a mail sorting system.

C. *Adaptive thresholding control technique*

The *bottleneck* of the overall system performance lies in the *accuracy* of the address similarity measures. Due to the unconstrained and omni-scriptor nature of the input, the cumulative errors resulting from the preprocessing of the handwritten address remain the major constraint to address field recognition.

This control strategy has two advantages. One is that it tolerates or compromises the propagation errors accumulated from earlier processes, such as address preprocessing and features extraction, by overestimating the possible postcodes so that the actual postcode tends not to be missed. The second is that buffering leads to schemes where evidence for the confidence of hypotheses and validity of the top hypothesis is successively fed forward through later steps in the decision rules for correct interpretation of the postcode. It is capable of altering the thresholding scheme to adapt to the ambiguous circumstances at the input stream of the neural network classifier by checking the validity of the top hypothesis. Therefore, this thresholding-control strategy is defined as *adaptive* to a noisy environment in this paper. The result is a more robust handwritten address verification process, since several rules may act in concert to increase or decrease the confidence of a candidate postcode rather than a single rule or algorithm as described elsewhere [12].

The adaptive thresholding-control strategy was accomplished through the following four units:-

i. A moderate thresholding scheme that uses low accept threshold T_{acc} ,

- It is capable of minimising lexicon holes that resulted from errors in the preprocessing and feature extraction. This increases the probability of finding the true address from the lexicon
- Property: Number of lexicon holes $\propto$ T_{acc}

ii. N-Buffering method that buffers the top N hypotheses for the postcodes that satisfy T_{acc} ,

- It is capable of minimising lexicon holes that resulted from errors in the neural network classifier
- Property: Number of lexicon holes $\propto$ $\frac{1}{N}$
- Buffering N hypotheses is essential because the first occurrence of a qualified event is not always correct

iii. Buffer Sorting

- It sorts the buffer according to the address confidence in an ascending order
- The hypothesis with maximum address confidence is selected

iv. Decision rules- to make a decision on the final interpretation of the postcode based on a set of rules

ACCEPT rules:

Rule 1: *the top hypothesis has the maximum address confidence in the buffer*
Rule 2: *the top hypothesis is valid*
Rule 3: *the address confidence of the next top valid hypothesis is greater than a threshold,* T_{acc}^{d}
Rule 4: *the difference of the maximum and next maximum address confidence exceeds a threshold value*

REJECT rules:
Rule 5: *the number of iterations in searching for N hypotheses that satisfy* T_{acc} *from the ranked hypotheses list exceeds a limit AND the buffer is empty*
Rule 6: *reaches end of file of the ranked hypotheses list AND the buffer is empty*

RECOVER rule:
Rule 7: *the address confidence exceeds a high recovery threshold level,* T_{rec}

The system uses *Rules* 1 to 7 to yield a highly accurate result of the postcode by producing one of the three *states*, ACCEPTED, REJECTED or RECOVERED at Output B (see Figure 3). The control structure is,

```
if (Rule 5 OR Rule 6)
    then
        Output Status = REJECTED;
    else
        if (Rule 1)
            then
                if (Rule 2)
                    then
                        Output Status = ACCEPTED;
                    else
                        if (Rule 3 OR Rule 4)
                            then
                                Output Status = ACCEPTED;
                                else
                                Output Status = REJECTED;
                        fi;
                fi;
            else
                if (Rule 7 OR Rule 4)
                    then
                        Output Status = RECOVERED;
                    else
                        Output Status = REJECTED;
                fi;
        fi;
fi;
```

where $T_{acc} < T_{acc}^{d} < T_{rec}$. A higher threshold, T_{acc}^{d} is used to prevent wrong classification in the case of high probability of confusion in the neural network classification and T_{rec} is used for postcode recovery in the case of *perfect* address matches.

IV. Experimental Results

The proposed method was trained using a training set of 450 handwritten addresses and tested using a test set of 1,071 handwritten addresses. In total, 450 + 1,071 (1521) fictitious but realistic addresses were randomly generated by computer. The handwritten addresses were partially constrained using boxes and horizontal guidelines for entering the postcode and the rest of the address. The colour of the boxes and horizontal guidelines were chosen such that they dropped-out during the scanning process. The addresses were then scanned using a Hewlett Packard Scanjet flat bed scanner operated in an automatic feeding mode at a fixed scan area of 6.0 cm 14.0 cm to simulate the image acquisition process at the Singapore Postal Centre (SPC). The binary images were scanned at a resolution of 200 dpi (dots per inch) in both x and y-axes for consistency with the resolution standards used on international postal sorting systems (e.g. Siemens sorting machines installed at SPC).

The addresses to use for evaluation were selected from a database consisting of around 0.2 million Singapore addresses using a randomised computer search.

They were printed out separately and grouped into sets. These sets of addresses were presented to the writers and each person was asked to write one set (3 addresses) on three address cards using their normal handwriting. Before writing, they were told to write the address using the guidelines and boxes on the card, in which the six-digit postcode should be written inside the boxes. The training and test databases were collected as follows:

The database was collected by asking over 360 different people from a random population were approached in several public places to write the three addresses. Some of them were also collected from university students on the campus hostels. Examples of the images used in the analysis are shown in Figure 7.

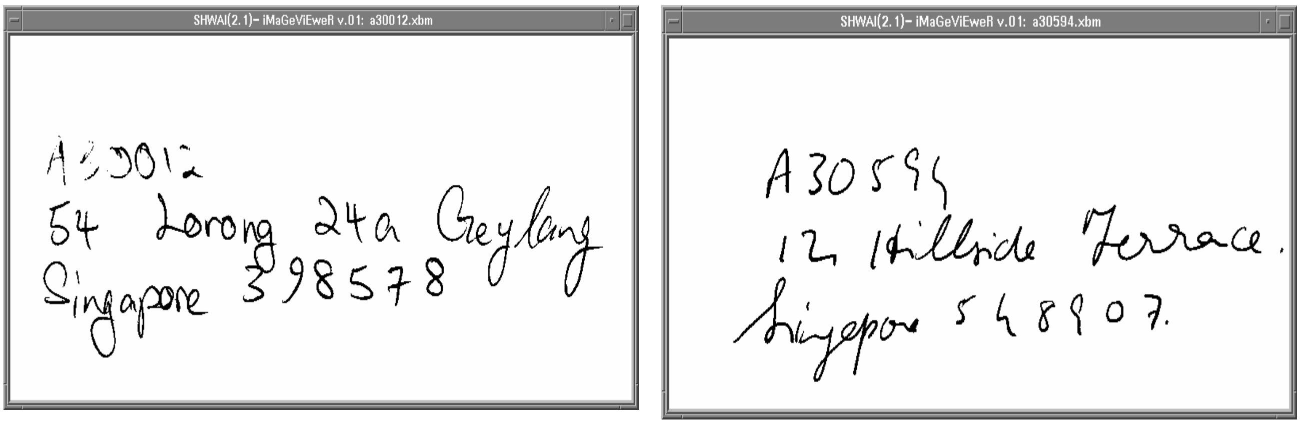

Figure 7. Examples of two address images used to evaluate the system.

The system performance was compared against the current Singapore Post system by running the 1,071 test addresses through their system.

Table 2 shows that the recognition rate of the proposed system on the test set was 63.7% with an error rate of 0.6%. The errors were mainly due to falsely recognised postcodes being verified and falsely recognised addresses being recovered by the system. The rejection was due to OCR/dictionary-lookup errors, no valid postcodes were found from the substitution sets by the dictionary lookup and non-OCR readable addresses. The experimental results showed significant reduction in the error rate compared to the error rate of 24.0% using OCR only to recognise the six digits postcode. This achievement is possible at the expense of more addresses being rejected since manual sorting is cheaper than to recover a wrongly destined letter. To compare the performance of the system researched and developed in this work with the performance obtained from the commercial system currently installed at the Singapore Post Centre, the final system was evaluated using the same test set. From Table 2, it can be seen that only 47.8% of the images were correctly recognised by the current OCR systems installed in the Singapore Post Centre. The major errors that occurred in the Singapore Post Centre system were classification errors and postcode location error (0.7%). The high rejection rate of 51.5% was due to the errors in the handwritten postcode recognition.

V. Conclusions

We have presented a design for conflict resolution in unconstrained Singapore handwritten address recognition for a real-time handwritten address interpretation application. The significant characteristics of the design are the use of contextual knowledge in the form of a postal directory and address database, lexicon holes reduction, and agreement for confidence combination for improved robustness. The combination of the decisions of an OCR algorithm and an expert system has proved to be a robust strategy for pattern classification of numerical strings especially in the presence of noise. An adaptive thresholding-control technique for the unconstrained handwritten address fields recognition was presented. It has proved to be more noise tolerant with the *N*-Buffering method especially when recognising poor quality handwriting that degrades Handwritten Word Model Recognition components such as the feature extractor. Thereby, it improves reliability. The performance of the system was evaluated in terms of the overall recognition rate, rejection rate and error rate. The goal of the effort was to achieve low error rates. The significant contribution of this expert system is a high error reduction rate of 97.5% for accurate sorting of the postcode. The performance is shown to be effective and viable for Singapore postal sorting and is adaptable to other applications.

	OCR only	Proposed System	Singapore Post System
Recognition Rate	71.2% (763)	63.7% (682)	47.8% (512)
Rejection Rate	4.8% (51)	35.7% (383)	51.5% (552)
Error Rate	24.0% (257)	0.6% (6)	0.7% (7)

Table 2: System performance comparison using 1,071 handwritten addresses for the recognition of 6-digit postcode

Acknowledgment

The authors would like to thank Koh Chin Peng, Chong Fon Lam and Lam Y. K. of Singapore Post Centre for assistance in obtaining the performance results using the Singapore Post OCR sorting equipment reported in this paper.

References

[1] R. M. Bozinovic and S. N. Srihari, “Off-line cursive script word recognition”, in: IEEE Trans. on Pattern Analysis and Machine Intelligence, 11(1), 1989, pp. 68-83.

[2] M. Cheriet, Y. S. Huang and C. Y. Suen, “Background region-based algorithm for the segmentation of connected digits”, in: Proc. of the Eleventh Int. Conf. on Pattern Recognition, The Hague, The Netherlands,1992, pp. 619-622.

[3] A. C. Downton and E. Kabir, “Verification techniques for high-performance OCR of hand printed postcodes”, in: IEE Colloquium on Character Recognition and Applications, London, UK, 7, 1989, pp. 1-9.

[4] N. F. Findler and J. V. Leewen, “A family of similarity measures between two strings”, in: IEEE Trans. on Pattern Analysis and Machine Intelligence, 1(1), 1979, pp. 116-118.

[5] P. Gader, J. Keller, and J. Cai, “A fuzzy logic system for the detection and recognition of street number fields on handwritten postal address”, in: IEEE Trans. on Fuzzy Systems, 3(1), 1995, pp. 83-95.

[6] D. Guillevic and C. Y. Suen, “Cursive script recognition: a fast reader scheme”, in: Proc. of the second Int. Conf. on Document Analysis and Recognition, Tsukuba Science City, Japan, 1993, pp. 311-314.

[7] R. M. Haralick, S.R. Sternberg and X. Zhuang, “Image analysis using mathematical morphology”, in: IEEE Trans. on Pattern Analysis and Machine Intelligence 9(4), 1987, pp. 532-550.

[8] Hendrawan, “Recognition and verification of handwritten postal addresses”, Ph.D. dissertation, Electronic Systems Engineering Dept., University of Essex, UK, 1994.

[9] A. K. Jain and B. Chandrasekaran, Dimensionality and sample size considerations in pattern recognition practice. In: Krishnaiah P R and Kanal L N, eds., Handbook of Statistics, 2, North-Holland, Amsterdam 1982.

[10] F. Kimura, M. Shridhar and Z. Chen, Improvements of a lexicon directed algorithm for recognition of unconstrained handwritten words. In: Proc. of the second Int. Conf. on Document Analysis and Reognition, Tsukuba Science City, Japan, 1993, pp. 18-22.

[11] C. K. Lee and G. Leedham, A Novel Fusion of Holistic and Analytical Paradigms for the Recognition of Handwritten Address Fields, in: Proc. 13th Int. Conf. on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, New Orleans, Louisiana, USA, 2000, pp. 492-501.

[12] S. Madhvanath, E. Kleinberg, and V. Govindaraju, Empirical design of a multi-classifier thresholding/control strategy for recognition of handwritten street names. in: Int. Journal of Pattern Recognition and Artificial Intelligence 11(6), 1997, pp. 933-946.

[13] B. T. Mitchell and A. M. Gillies, A model-based computer vision system for recognising handwritten zip codes, in: Machine Vision and Applications 2, 1989, pp. 231-243.

[14] J. V. Moreau, A new system for automatic reading of postal checks, In: Impedovo S and Simon J C (eds.), From Pixels to Features III, Elsevier, 1992.

[15] S. N. Srihari, E. Cohen, J. J. Hull, and L. Kuan, A system to locate and recognise zip codes in handwritten addresses, in Int. Journal of Research and Engineering- Postal Applications 1, 1989, pp. 37-45.

[16] S. N. Srihari, and E. J. Kuebert, Integration of hand-written address interpretation technology into United States postal service remote computer reader system, in: Proc. of the Fourth Int. Conf. on Document Analysis and Recognition, Ulm, Germany, 2, 1997, pp. 892-896.

[17] R. W. S. Tregidgo and A. C. Downton, High performance character recognition for off-line handwritten British postcodes, in: IEE Colloquium on Binary Image Processing, Technology and Application, 6, 1991, pp. 1-5.

GRAHAM LEEDHAM graduated in Electrical and Electronic Engineering from Leeds University, UK and obtained his MSc and PhD in Electronics from Southampton University, UK. He was a member of academic staff of Essex University (UK) from 1984 to 1994 and has held visiting research positions in the USA, Canada and Australia. His research interests are in real-time image processing and pattern recognition with particular interest in the analysis and processing of handwriting. He has published over 130 refereed research papers in these areas and organised, chaired or served on the committees of many international conferences. Dr Leedham is a Fellow of the Institution of Electrical Engineers and is Associate Professor and Head, Division of Computing Systems in the School of Computer Engineering at Nanyang Technological University, Singapore.

CHIN KEONG LEE graduated in Computer Engineering from Nanyang Technological University, Singapore and obtained his Bachelor and PhD in Computer Engineering. He was conferred a PhD in Computer Engineering in June 2001. His research interests are in handwriting image processing and pattern recognition with particular interest in the analysis and verification of handwritten postal addresses. He has published 10 research papers in international journals, conferences, school research magazine and local newspaper. Dr Lee is currently a technical manager for software simulation research in fixed wireless communication.