A SYSTEM FOR RECOGNITION OF DESTINATION ADDRESS IN POSTAL DOCUMENTS OF INDIA

Ramit Kumar Roy[1*], Umapada Pal[2], Kaushik Roy[3] and Fumitaka Kimura[4]

[1]St.Xavier's College (Autonomous), Kolkata, India

[2]CVPR Unit, ISI, Kolkata, India

[3]Dept. of Comp Sc., WBSU, Barasat, India

[4]Graduate School of Eng., Mie University, Japan

Email: ramitkumar.roy@sxccal.edu[1]*(corresponding author), umapada_pal@yahoo.com[2], kaushik.mrg@gmail.com[3], kimura@hi.info.mie-u.ac.jp[4]

DOI: https://doi.org/10.22452/mjcs.vol33no3.3

ABSTRACT

Recognition of destination address is compulsory for automation of the postal system in India. Our observation found that such recognition becomes a very challenging task due to inter-mixing of three languages (Hindi, English and the official language of the particular state in which the postal document is supposed to reach).In this paper, our attempt towards development of a dynamic programming based system for city-name and pin code recognition of destination address in postal documents of India not only managed to address the difficulties related to identification of the scripts but also managed to get rid of those problems which is generated due to character touching in postal documents. For city-name recognition, lexicon information is used. However, no lexicon information is used for pin code recognition since an Indian pin code contains only 6 digits. We obtained 99.55% reliability from tri-lingual city-name recognition system where error rates are 0.20% and rejection rates are 28.11%. From our experiment on recognition of handwritten pin codes, 99.01% reliability rate, 0.83% error rate and15.27% rejection rate are obtained. Furthermore, to enhance the city-name results by distributing the lexicon size district-wise, we conducted an experiment and presented the results.

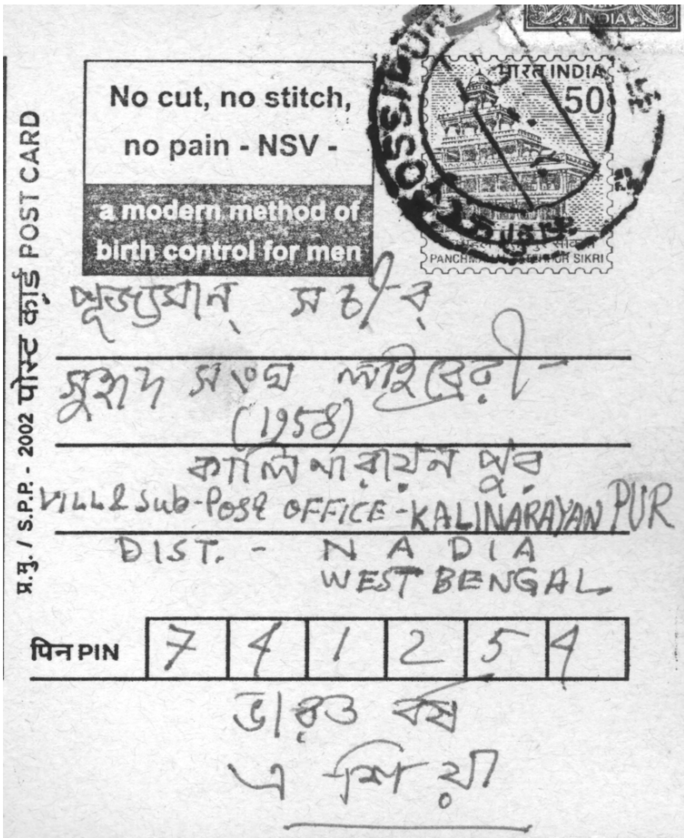
Keywords: ***Destination Address, Indian script recognition, Indian Pin code recognition, Document analysis, Lexicon information***

1.0 INTRODUCTION

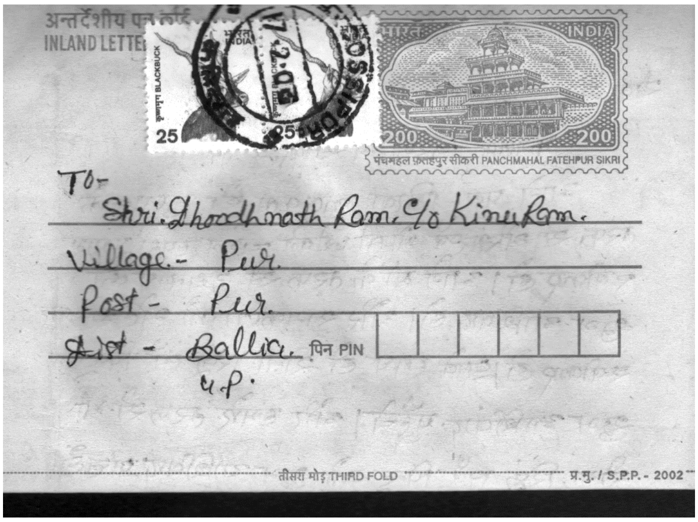
Nowadays, because of online purchase through Amazon, flip kart etc., number of postal documents is increasing day by day. In fact, number of parcels has increased tremendously for the past 5 years. Thus, automatic postal sorting system is very much necessary for handling such huge number of postal document and to reduce the manpower effort of human sorting. Automatic recognition of the destination address of India is a very challenging task due to multi-lingual and multi-script behavior. For the postal system of developed countries, there exist many published research papers dealing with automation of postal documents written in non-Indian languages[1,2,11,12,13,14,24,25,27,28]. Most of the developed countries consider only a single script; hence, to develop postal system for such countries is easy. However, India is multi-lingual and multi-script country where postal document may contain any of the languages. Thus, postal system development for India is a difficult task. Because of such complexity, only a few published research papers can be found for the automation of postal documents written in Indian languages [3,4,8,26]. In addition, only three or four postal sorting machines are available in India and it works only for English and such system could not handle postal document written in Indian scripts.

Development of a recognition system of destination address for a country like India is a very challenging task. In India, there are twenty three (23) official languages and eleven (11) scripts that are used at the time for writing these 23 languages [17]. However, in spite of the fact that we have so many languages and so many scripts in India, it has been observed that the destination address (the address where a particular postal document is expected to finally reach) is often written using only 3 languages: (1) English; (2) Hindi; and(3)local state language. Therefore, it is needless to mention that any document which is used for the purpose of postal communication in India is tri-lingual, as it often comprises these three languages. For our work, we prepared a database for which we had acquired as many as 7500 documents exclusively utilized for postal purposes. These documents are obtained from an Indian state named West Bengal. The statistical details regarding our database are available in[4]. From our statistical computation and analysis, we observed the following: (1) 76.32% of the postal documents are written in English; (2) Bangla (Bengali) is used to write 12.37% of the postal documents; and (3) Hindi (Devnagari) is deployed while writing 10.21% of the documents used for postal purposes.

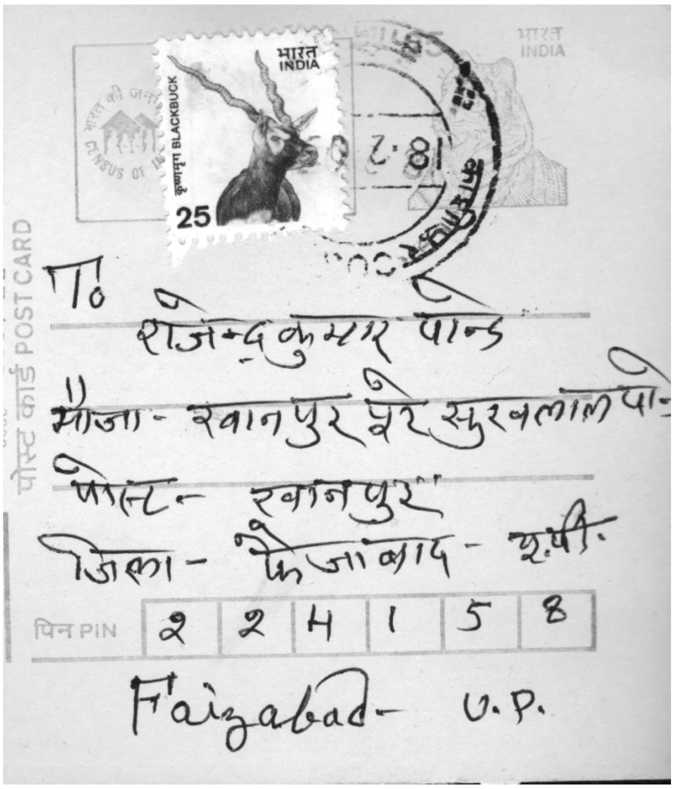
Based on the tri-lingual nature of those documents, we propose a tri-lingual city-name and pin code (PIN) recognition system for recognition of the destination address of Indian postal documents. By combining both the results of recognition of pin codes and recognition of city-names, reliability in the existing Indian postal automation system can be enhanced.



(a)



(b)



(c)

Fig.1:Examples of some Indian postal documents. (a) Here the city name is written in both Bangla and English whereas pin-code is written in English only (b) whole postal document is written in English, and it has no pin-code (c) the name of the city is written is Hindi and English, but the pin-code is written in English only.

Generally, we implement any of the two approaches as discussed here for multi-script OCR: (1) The script is identified at the very beginning and accordingly we use the proper OCR which is based on the identified script; (2) A system is developed so that it is capable of recognizing all the scripts' samples itself. Currently, a number of studies related to the identification of scripts have already been published [3,16,22]. Since in most cases, it has been observed that destination address block in India contains more than two scripts, hence accurate identification of the script from such an address block becomes very difficult and challenging.

In Fig.1(a) the name of the city is written in Bangla as well as in English whereas pin-code is written in English only. From Fig.1(b) we can observe that whole postal document is written in English only but there is no pin-code. In Fig.1(c) the name of the city is written is Hindi and English, but the pin-code is written in English only. From the database we have seen that one single line of the destination block of address may be a combination of two or more than two scripts (see Fig.1(a)). Also, in case of many of the Indian postal documents, characters in a city-name or the digits using which the PIN is written are often found touching each other. Two digits, three digits, four digits or even five digits touching strings are very common in case of Indian pin codes. Examples of touching strings of different digits of Indian pin codes written in Devnagari are shown below in Fig.2. Accurately segmenting such touching strings is very challenging.

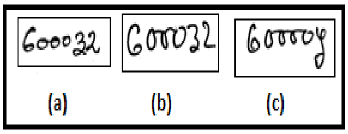


Fig.2 Examples of touching digits of Indian PIN codes written in Hindi. (a) Touching of two digits with one another. (b) Touching of three digits with one another. (c) Touching of four digits with one another

Taking into consideration all these perspectives, through our paper, a scheme for segmentation as well as a scheme for recognition of destination address for Indian postal documents is proposed. The scheme is as follows: First, binarization of the input string is done, which is followed by the slant correction of that particular input string. The binarized input strings are then segmented into primitive characters. By primitive characters, we mean individual character/numeral or its parts. Such segmentation is done through the implementation of water reservoir principle as discussed in [9].Ideally, every primitive consists of one single digit/character or a sub-image of one single digit/character. Dynamic programming (DP) is deployed for merging of the primitive components which are present in a string into characters/digits, and to get the best segmentation. DP uses the total likelihood of characters as an objective function. To determine the presence of likelihood in a character or to determine the presence of likelihood in a digit, we implemented Modified Quadratic Discriminant Function(MQDF)on the basis of the directional features of the various contour points in the primitive components.

Fig. 3 presents the system that we proposed for recognizing the names of the cities. Please note that our pin code recognition is similar to city-name recognition system but we do not require any lexicon. Based on the diagram, we use city name lexicon which is the vocabulary of city names.

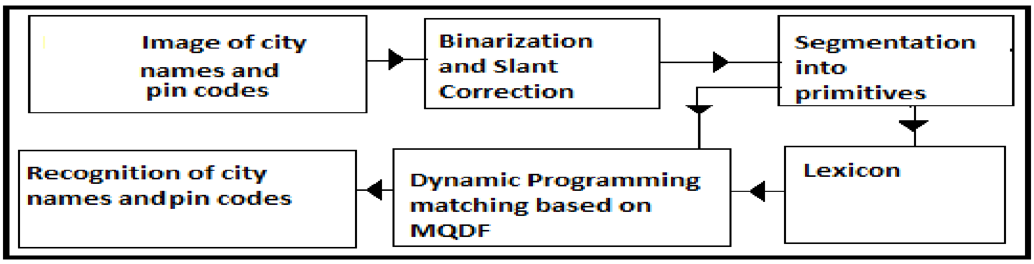


Fig.3: Block diagram showing our proposed city-name recognition system.

2.0 A BRIEF REPORT ON INDIAN POSTAL SYSTEM

The Indian Postal department has more than 1,54,000 post offices spread across the country. On an average, a post office covers an area of more than 21sq. km, and it serves to an average population of over 8,000 people. This is the most widely spread postal system globally. In terms of its postal services, India has been divided into regions and each region in turn is under the leadership of a Postmaster General. Regions are again further sub-divided into sub-regions. A previous study in [4] had discussed the representation of first 2 digits of the PIN.A wide variety of postal documents such as post-cards, inland letters, aerogram, ordinary envelopes and special envelopes exist in India, and these postal documents are available in the post offices. The PIN which is always numerical and consists of six digits compulsorily is supposed to be written only in the six printed boxes exclusively made for that purpose in the postal documents. However, in many cases, we found that some users wrote only a part of the PIN and not the whole PIN such as *Kolkata-29* or *Kol-29* instead of*Kolkata-700029*.In these cases where ordinary envelopes are used for postal purposes, there are no printed boxes for writing the PIN. Moreover, we found that when the PIN is not written, it is required to recognize both the names of the city as well as the PIN. Therefore, we can conclude that the recognition of destination address is definitely not an easy task.

3.0 CORRECTION OF SLANTS AND PRE-SEGMENTATION OF CHARACTERS

3.1 Correction of Slants

Slant correction is a vital step in preprocessing in case of recognition of handwriting. This section of our work deals with the following two aspects chronologically: 1)Binarization of our input images; and 2)The images which are first binarized, and then subjected to correction of slants. We followed the procedure as described in [5] where we first need to binarize the input images. We estimated the slants of our input images based on the information presented in the input images' contour. We also followed the procedure as described in [7] where we performed the correction of slants of the input images which were already binarized by us.

3.2 Segmentation of primitives

For optimal character segmentation using DP, input string is segmented to get primitives. A city-name or pin code may sometimes touch and create a big cavity portion because of such touching. Using profile behavior and water reservoir properties such cavity regions may be obtained and the deep most point of each cavity part is treated as pre-segmentation point. To remove some improper pre-segmentation points, the median (M) of the heights of different cavity regions is computed and the cavities with height greater than ($M \times 0.8$)are treated for segmentation. This is done to consider the small cavities. Our aim in pre-segmentation is to divide a city-name or a pin code into individual characters so that over segmentation can be avoided. In Fig.4(b),the lower cavity parts for a Bangla city-name of Fig.4(a) are shown. An image of a city-name is split into segments at each pre-segmentation column and connected components are analyzed to get an individual component. In Fig 4(c) connected components as well as their enclosing boxes are shown. These boxes are then numbered as shown at upper side of Fig.4(c). Such connected components are the primitive segments. A full character or part of a character can be a primitive.

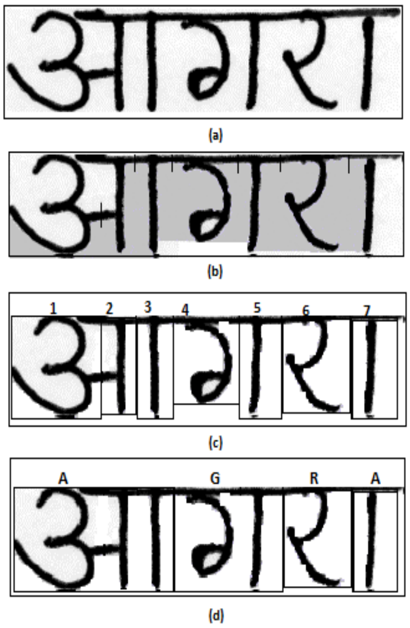


Fig. 4:(a).An example of pre-segmentation of the input city-name AGRA(आगरा). (b) Reservoirs at the bottom (indicated using grey colour). The pre-segmentation columns have been shown by drawing small lines vertically.(c) Individually segmented primitives have been indicated using boxes those are disjoint and are accordingly numbered. (d) Optimum segmented characters with their respective codes are shown.

4.0 EXTRACTION OF FEATURES

The histograms of the direction chain code in the contour points of the components are taken into account as the recognition features [6].For the purpose of recognition, 64 dimensional direction chain code features are used. The details of feature extraction procedure are given as follows. First, the bounding box is divided into 7 × 7 blocks and for each block, the direction chain code for every contour point is obtained. Then, the frequency of direction codes is noted. We use chain code in four directions that in horizontal, slanted with 45 degree, vertical and slanted with135 degree. Thus, for each block, an array of 4integer values representing the frequencies of chain code in these four directions is obtained and used as feature in our scheme. Since we got 4 features for each block, we got 196(7 × 7 × 4) features for 7 × 7 blocks. Using a Gaussian filter, we manage to reduce the feature dimension from7 × 7 blocks to 4 × 4 blocks, thus we manage to obtain 64 (4 × 4 × 4) dimensional features. To make it independent of size, the feature vector is normalized by dividing each component by the character height.

Speed is one of the important issues for feature extraction. The histogram of cumulative orientation based on a study by [7] is computed and used to get faster feature extraction.

5.0 DYNAMIC PROGRAMMINGBASED SEGMENTATION AND RECOGNITION

Dynamic Programming (DP) is an optimization technique based on recursion. Since it is very difficult for proper character segmentation from input city name images, we propose dynamic programming based technique in our work so that we can take the best results from the input city name. In our work, the basis of a dynamic programming algorithm is a module which:(i) accepts a city-name as input image; (ii) has a string which is used to include the contextual information in it; (iii)has also a list of the primitives from the images of the names of the city; and (iv) returns a value representing the confidence that the image of the name of the city represents the string. For a given lexicon name of the city, the primitive segments of the city name images are matched and then merged with the characters in the lexicon of the city names in order to maximize the average likelihood of characters using dynamic programming. Suppose, Z primitive segments are present in an input city-name and in turn when such a name of the city is matched against any lexicon word comprising P characters, in that case a character may have in it a maximum of (1+Z-P) primitive segments.

5.1 Markov chain representation

We noted that primitive number of an image is about 2 times than the image character number. For optimum segmentation, DP is used using the total likelihood of characters as objective function [7]. The lexicon of city-names is used in DP for the use of contextual information. MQDF is used to calculate the likelihood of a component segment. For the use of DP, the boxes are sorted from left to right and top to bottom using the information of their centroid position. When the same x coordinate values are obtained for two or more boxes, we use top to bottom information for their sorting. In Fig.4(c),the numbers at the top of the boxes denote the order of the sorted boxes. It can be easily seen that, in most of the cases, the disjoint box segmentation and the box sorting process turns the segmentation problem to a simple Markov process. From Fig.4, it can be seen that the first three boxes correspond to alphabet “A” of the Devnagari city-name AGRA (आगरा), next two boxes correspond to character “G”, box 6 corresponds to character “R”, and the last box 7 corresponds to alphabet “A”. Character codes of this city-name image are shown in Fig. 4(d).Such assignment is also mentioned as follows to ensure better clarity. For example, in case of the word AGRA,

	A	G	R	A
i -->	1	2	3	4
$j(i)$ -->	3	5	6	7

where i is the letter number, and $j(i)$ is the number of the last box that corresponds to the i-th letter. It can be seen that the number of the box that corresponds to the i-th letter is $j(-1+i)+1$.

5.2 MQDF for likelihood of characters

MQDF or Modified Quadratic Discriminant Function is a very user-friendly and highly powerful classifier [23].

Mathematical equation of MQDF is as follows.

$$g(X)=\{|X-\hat{M}|^2-\sum_{i=1}^{k}\frac{\lambda_i}{\lambda_i+h^2}[\phi_i^T(X-\hat{M})]^2\}/h^2 + \ln[h^{2(n-k)}\prod_{i=1}^{k}(\lambda_i+h^2)] \qquad \text{----(5)}$$

where the input feature vector is denoted by X, the sample mean vector for a character class is denoted by $\hat{M}$, eigen values and eigenvectors of the sample covariance matrix are denoted by λ_i and ϕ_i. $h^2$and k are two constant parameters whose value are selected through experiment to get the best accuracy. In our experiment, we set k as 20 and h^2as $3/8*\sigma^2$. Here σ^2 denotes the mean of eigen values $\lambda_{i's}$ over the value i and the character classes.

Given a feature vector as computed above for a character class, $g(X)$ is calculated for the character class specified by a city name lexicon and based on this $g(X)$, recognition can be done.

6.0 RESULTS AND DISCUSSION

6.1 Data details

In this study, we procured more than 16000 samples of handwritten data. The collection comprises4257 handwritten city-name samples written in Hindi, 8625 handwritten city-name samples written in Bangla and 3250 handwritten city-name samples written in English. There are a total of 290 city-name classes where each class has at least 20 samples. Table 1presents the classes of city-names and the number of characters found on an average in handwritten city-names.

Table 1**:** Classes of city-names and average length of the handwritten city-names.

City-name classes	Average length of handwritten city-names (This is obtained by taking into account the number of characters present)
117 for Hindi	6.86
84 in case of Bangla	5.85
89 in case of English	8.46

Handwritten city-names had been collected not only from a wide variety of postal documents such as post-cards, inland letters, aerogram, ordinary envelopes and special envelopes which exist in India but also from the feedback given by people using our specially designed forms. The sample of one of the forms is shown in Fig.5.For the PIN recognition, we collected a total of 16300 data and details of the collections are discussed in [21].

Rajgir	Rajgir	Srinagar	Srinagar
Delhi	Delhi	Kargil	Kargil
Agra	Agra	Amritsar	Amritsar
Jaipur	Jaipur	Jalandhar	Jalandhar
Jodhpur	Jodhpur	Ludhiana	Ludhiana
Jaisalmer	Jaisalmer	Patiala	Patiala
Bikaner	Bikaner	Dehradun	Dehradun
Barmer	Barmer	Mussoorie	Mussoorie

Fig.5:Sampleofspecially designed form used for our data collection

6.2 Result computation measures

To compute our results, we used different kinds of measures such as rate of recognition, rate of error, rate of rejection, and reliability. The calculations of these measures are defined as follows. Let N_C, N_E and N_R be the number of correctly classified, misclassified, and rejected city name, respectively. Also let N_T be the total number of city names tested, thus $N_T= (N_C +N_E +N_R)$. Under this notation, by recognition rate we mean $(N_C*100)/N_T$, by error rate we mean $(N_E*100)/N_T$, by rejection rate we mean $(N_R*100)/N_T$ and by reliability we mean $(N_C*100) / (N_E+N_C)$. Using these definitions, accuracies provided in different tables are calculated.

6.3 Results On City-Name Recognition

6.3.1 Results of global tri-lingual recognition city-names

From our result, we concluded that the overall percentage of accuracy for handwritten names of cities which were written using more than one scripts was 92.25%, without considering any rejection. We also found that the overall accuracy of 95.55% and 96.36% was achieved by considering the first three and the five top choices of the recognition results, respectively. Detailed experimental results taking into account the three scripts in recognizing city-names which were written using more than one script as well as the various top choices are presented in Table 2. Based on the results, we concluded that samples of city-names in Hindi show low results in comparison to samples of city-names in Bangla and samples of city-names in English. Such lower results in Hindi are due to the presence of compound characters and also because of shape similarity of some characters in Hindi. For the first top choice, it can be seen from the table that Bangla city name gives more than 3% better results than Hindi and more than2% better results than English. The accuracy of our trilingual city-name recognition system based on different choices is depicted in the graph shown in Fig.6.

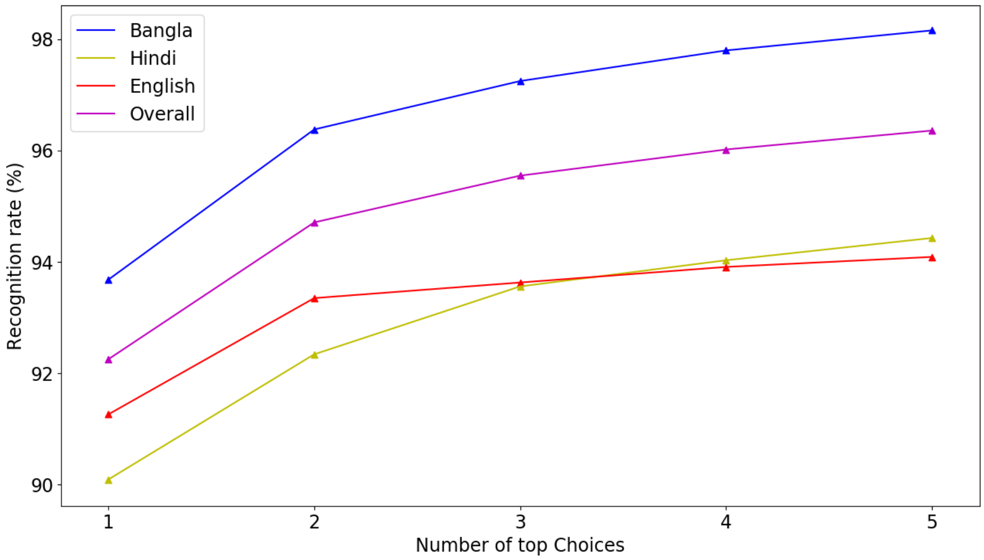


Fig.6: Accuracy of our tri-lingual city-name recognition system based on different choices

Table 2: Recognition results of tri-lingual city-names on the basis of various top choices of names of cities written in Bangla, Hindi and English (here 0.0% is the rate of rejection).

Number of top Choice(s)	Rate (%) of recognition			
	Bangla	Hindi	English	Overall
1	93.68	90.09	91.26	92.25
2	96.38	92.34	93.35	94.71
3	97.25	93.56	93.63	95.55
4	97.80	94.03	93.91	96.02
5	98.16	94.43	94.09	96.36

6.3.2 Recognition of results for mono-lingual names of the city

To analyze the results of city-names, which are mono-lingual in nature, we calculated the results of recognition. The percentage of accuracy we achieved are 94.08, 90.16 and 91.63 respectively by taking into account samples of individual scripts separately. Table 3displays the results of city-names which are mono-lingual in nature. The results are from the various top choices of names of the cities written in the three languages viz. Bangla, Hindi and English. From the two tables i.e. Table 2 and Table 3, we can conclude that both the workings of the names of the cities which are by nature tri-lingual and mono-lingual are at par .As for tri-lingual city name recognition, we can also see the lower results in Hindi in mono-lingual due to the presence of compound characters and also the similarity of shapes of some characters in Hindi. For the first top choices, it can be seen from Table 3 that Bangla city name gives more than 4% better results than Hindi and more that 2% better results than English. Accuracy of our mono-lingual city-name recognition system based on different choices is depicted in Fig.7.

6.3.3 Results of rejection versus Results of reliability

We had made necessary calculations in order to determine the results of rejection versus the results of reliability in case of recognition of tri-lingual handwritten city-names. We found that our proposed system ensures a reliability of 99.55% where error and rejection rates are0.20% and 28.11%, respectively. The reliability of recognition of city-names with different rates of rejection is discussed in Table 4. Criteria for rejection are:(i) value of optimal likelihood of the names of the city which is recognized as the best; and (ii) the difference of the values of optimal likelihood of the city name which were recognized as the best and the second best.

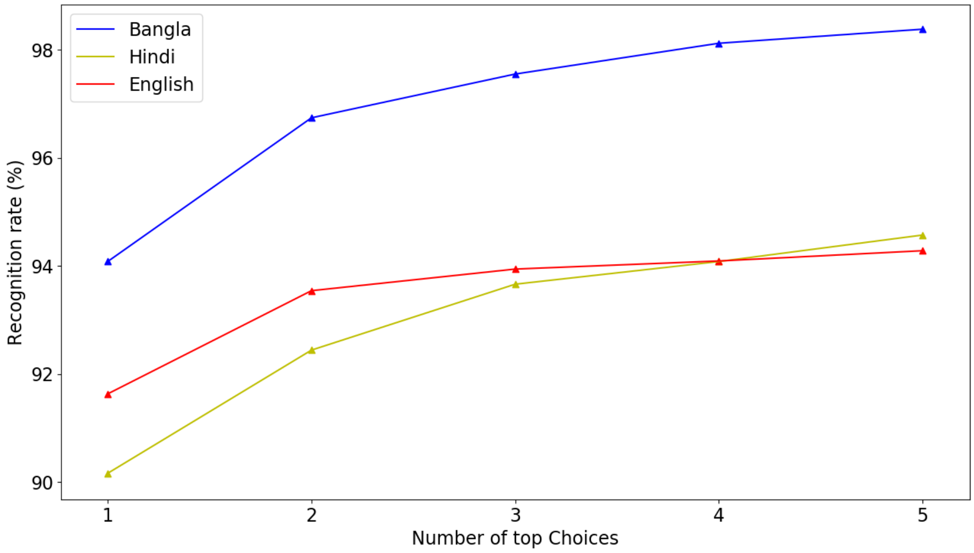


Fig.7:Accuracy of our mono-lingual city-name recognition system based on different choices.

Table 3: Results of names of cities which are mono-lingual in nature on the basis of various top choices in names of the cities written in Bangla, Hindi, and English (considering 0.0% rejection)

Choice(s)	Rate (%) of recognition		
	Bangla	Hindi	English
Top 1	94.08	90.16	91.63
Top 2	96.74	92.44	93.54
Top 3	97.55	93.66	93.94
Top 4	98.12	94.08	94.09
Top 5	98.38	94.57	94.28

Table4: Results of Reliability(%) and Error (%) in our work and Rate of Rejection (%)

Percentage (%) of reliability	Percentage (%) of error	Rate (%) of rejection
95.04%	4.62%	4.44%
98.00%	1.65%	11.69%
99.04%	0.67%	18.01%
99.55%	0.20%	28.11%

6.3.4 A Comparative Study of Results

To our knowledge, this is the first work on tri-lingual city name (written in English or Hindi or Bangla) recognition, hence, we could not compare our results with any previous works. However, for mono-lingual cases, we found only a single work on recognizing words written in Hindi which used HMM on a small class of Hindi words dataset [10]. The work deals with only 50 classes of words written in Hindi where an accuracy level of 82.89% was achievedusing3000 words for testing purposes. However, we are able to achieve a better result with90.16 % in recognition of city-names written in Hindi using 117 classes. We firmly believe that the results we obtained on recognition of names of the cities for mono-lingual as well as for tri-lingual will be of a great help to the researchers to get the state of the art results towards this work.

6.3.5 Analysis of errors

While executing our research work, our findings revealed that in those situations where points of pre-segmentation which we obtained from the module of pre-segmentation did not have the real points of segmentation of characters. Similarity in the shape of some of the names of the cities was another major reason behind the miss-recognition. From our research work, we found that for city-names which were written in Hindi, we obtained the lowest results. One most important factor for not having higher rates of recognition was the presence of compound characters which in turn had made the structure of the city-names written in Hindi, a highly complex one. In Hindi, there are more than 200 compound characters but we found it very difficult to get the real samples of such a large number of compound characters to train our system so that their likelihood can be computed. To obtain higher results, we plan to collect more samples of compound characters in the future.

6.4 Results On PIN Recognition

6.4.1 Results of recognition of PIN

Our present research work, which deals with recognizing strings of hand-written PIN has achieved an accuracy of 94.14%, taking into consideration a rejection rate of 0.15%.By taking into consideration the first two (three) choices of the results of recognition, an overall accuracy that we have achievedis96.26% (96.68%).Table 5 as shown below deals with the detailed results of the three scripts of our multi-script PIN along with the various choices. From the table, we found that there is an increase in recognition accuracy by 2.70% (96.02% instead of 93.32%) when we took into account 2(two) top choices and not 1(one) top choice as in Devnagari. Our analysis of results revealed that the enhancement of this 2.70% in accuracy is mainly because there are some digits which are similar in shape and that in turn has made the two strings of PIN look the same in appearance.

Table 5:Recognition of PIN based on top choices (Here 0.15% is the percentage of rejection)

Choice(s)	Percentage(%) of recognition		
	Bangla	Devnagari	English
Top 1	92.24	93.32	95.27
Top 2	94.73	96.02	96.92
Top 3	95.47	96.53	97.18

6.4.2 Rate of Rejection versus rate of error

To compute the reliability of our proposed system, we carried out necessary experiments. From the results of these experiments, we achieved a reliability of 99.01% where the rate of error was 0.83%. The rejection rate is 15.27% where it was done on the basis of: (i) the value of optimal likelihood of the pin code which was recognized as the best; and (ii) the difference of values of optimal likelihood as found in the pin code which were recognized as the best and the second best.

6.4.3 Comparison of results

We are unable to compare our results with the existing works as we could not find any published research work dealing with recognition of handwritten multi-script (Bangla, Devnagari and English) pin codes. However, there exists some research works on recognition of handwritten isolated digits and the comparative results are presented in Table 6. From the results, it shows that our method performs better than the existing methods.

Table 6: Results on isolated digits.

Name of the script	Methods	Size of data	Accuracy percentage
Bangla	Refer [15]	16K	95.05%
	Refer [16]	14K+650	96.66%
	Our work when isolated pin codes are considered	16128	98.10%
Devnagari	Bhattacharya et al. [19]	22535	92.83%
	Hanmandlu and Murthy [18]	Not known	95.00%
	Proposed method when isolated pin code are considered	23340	98.41%

6.4.4 Analysis of Errors

This section deals with the classification of different recognition errors found in pin codes recognition. Table 7shows the distribution of errors for pin codes recognition while dealing with errors found in number of digits of the pin code.

Table 7: Distribution of errors in different digits.

Errors found in number of digits	Recognition errors
1	4.10%
2	0.85%
3	0.36%
4 or more digits	0.40%

6.5 Improvement of City-Name Recognition Based on Pin Information

The current postal system of India is the most extensively spread postal system in the world. To get better city-name recognition results of such a distributed post office system, we proposed to use pin code information. Since we used lexicon for recognition of city-names, we performed an experiment to determine the reduction of performance when the size of the lexicon is increased. This experiment was done for Bangla city-names. Table 8 shows the accuracy of different lexicon sizes. From the table, we can conclude that the level of accuracy reduces by about 5.89% (98.27%-92.38%) when lexicon size increases from 10 to 200.

Table 8: Accuracy on different lexicon sizes.

Diff. top choices	Lexicon size					
	10	20	50	100	150	200
1	98.27%	97.53%	95.41%	93.75%	93.09%	92.38%
2	99.22%	98.74%	97.65%	96.46%	95.93%	95.46%
3	99.50%	99.20%	98.28%	97.43%	96.92%	96.57%
4	99.57%	99.39%	98.68%	97.98%	97.52%	97.14%
5	99.69%	99.42%	98.88%	98.25%	97.89%	97.57%

Based on these results, we divided the lexicon of the pin code of a district and we consider all the city-names of a district in a single lexicon. For example, one of the biggest districts of West Bengal is 'Burdwan' and there are 32 city-names. These city-names are considered as the name of police stations. Thus, we take into consideration all the 32 city-names in a single lexicon and we, in turn, use this lexicon for the city-name recognition when the pin code is recognized as the pin code of 'Burdwan' district. Similarly, another district of West Bengal is 'Bankura' and there are 22 city-names. We consider all the 22 city-names in a single lexicon again and we use this lexicon for the city-name recognition when the pin code is recognized as the pin code of 'Bankura' district. From our experiment of the city-name lexicon distribution based on different districts, we obtained an average 2.53% better accuracy in tri-lingual city-name recognition.

7.0 CONCLUSION

Because of multi-lingual and multi-script behavior, system development towards Indian postal automation is very complex. Most of the existing systems deal with only maximum two languages/scripts. In this work, we presented a system for recognition of destination address in tri-lingual scenario for postal documents of India. Currently, there is only one work in the literature which deals with recognition of tri-lingual city-names [20]. However, there are some research works on isolated digits, but only a few research papers on recognition of Indian Pin numbers have been published. From the comparative results given in this paper, it can be seen that our results outperform the others. Our future plan will be to improve the pin code recognition results in order to achieve better accuracy using deep learning.

REFERENCES

[1] L. Liu and M. Koga and H. Fujisawa, "Lexicon driven segmentation and recognition of handwritten character strings for Japanese address reading", IEEE Trans. PAMI, vol. 24, pp. 1425-1437, 2002.

[2] R. Plamondon and S. N. Srihari, "On-Line and off-line handwritten recognition: A comprehensive survey", IEEE Trans. PAMI, vol.22, pp.62-84, 2000.

[3] U. Pal, K. Roy and F. Kimura, "A Lexicon Driven Handwritten City-name Recognition Scheme for Indian Postal Automation", IEICI Trans. on Inf. and Syst., vol.E92-D, No.5, pp. 1146-1158, 2009.

[4] S. Vajda, K. Roy, U. Pal, B. B. Chaudhuri and A. Belaid, "Automation of Indian Postal Documents written in Bangla and English", Int. Journal of PRAI, vol. 23, No. 8, pp. 1599–1632, 2009.

[5] N. Otsu, "A threshold selection method from gray-level histograms", IEEE Trans. SMC, vol.9, pp.62-66, 1979.

[6] F. Kimura, Y. Miyake and M. Sridhar, "Handwritten ZIP code recognition using Lexicon free word recognition algorithm", ICDAR, pp. 906-910, 1995.

[7] F. Kimura, S. Tsuruoka, Y. Miyke and M. Sridhar, "A Lexicon Directed Algorithm for Recognition of Unconstrained Handwritten Words", IEICE Trans. Inf. and Syst.,vol.E7-D, No.7, pp.785-793, 1994.

[8] U. Pal, R. K. Roy and F. Kimura, "Bangla and English City-name Recognition for Indian Postal Automation", ICPR, pp. 1985-1988, 2010.

[9] U. Pal, A. Belaid and Ch. Choisy, "Touching numeral segmentation using water reservoir concept", Pattern Recognition Letters, vol. 24, no. (1-3), pp. 261-272, 2003.

[10] S. K. Parui and B. Shaw, "Offline Handwritten Devnagari Word Recognition: An HMM Based Approach", PReMI 2007, pp. 528-535. 2007.

[11] U. Mahadevan, and S. N. Srihari, "Parsing and Recognition of City, State, and ZIP Codes in Handwritten Addresses", ICDAR, pp. 325-328, 1999.

[12] X. Wang, and T. Tsutsumida, "A New Method of Character Line Extraction from Mixed-unformatted Document Image for Japanese Mail Address Recognition", ICDAR, pp. 769-772, 1999.

[13] S. N. Srihari, and E .J. Keubert, "Integration of Hand-Written Address Interpretation Technology into the United States Postal Service Remote Computer Reader System", ICDAR, pp. 892-896. 1997.

[14] A. Kornai, "An Experimental HMM-Based Postal OCR System", ICASSP'97, pp. 3177-3180, 1997.

[15] Y. Wen, Y. Lu and P. Shi, "Handwritten Bangla digit recognition system and its application to postal automation", Pattern Recognition, vol.40, pp.99-107, 2007.

[16] K. Roy, "On the development of an optical character recognition system for Indian postal automation", Ph.D. Thesis, J.U., 2008.

[17] U. Pal, "Automatic Script Identification: A Survey", Vivek, vol.16, pp.26-35, 2006.

[18] M. Hanmandlu and O. V. Ramana Murthy, "Fuzzy model based recognition of handwritten numerals", Pattern Recognition, vol.40, pp. 1840-1854, 2007.

[19] U. Bhattacharya et al., "Neural combination of ANN and HMM for handwritten Devnagari numeral recognition", IWFHR, pp.613-618, 2006.

[20] U. Pal, R. K. Roy and F. Kimura, "Multi-lingual City-name Recognition for Indian Postal Automation", ICFHR, pp. 169-173, 2012.

[21] U. Pal, R. K. Roy, K. Roy and F. Kimura, "Indian Multi-Script Full Pin code String Recognition for Postal Automation", ICDAR,pp. 456-460,2009.

[22] Sk Md Obaidullah, Chayan Halder, K. C. Santosh, Nibaran Das and Kaushik Roy, "PHDIndic_11: page-level handwritten document image dataset of 11 official Indic scripts for script identification" Multimedia Tools Appl. 77(2): pp. 1643-1678, 2018.

[23] Fumitaka Kimura, Kenji Takashina, Shinji Tsuruoka, Yasuji Miyake, " Modified Quadratic Discriminant Functions and the Application to Chinese Character Recognition". IEEE Trans. PAMI, 9(1), pp. 149-153. 1987.

[24] Yawei Hou and Huailin Zhao, "Handwritten Digit Recognition Based on Depth Neural Network", ICIIBMS 2017, Track2: Artificial Intelligence, Robotics and Human-Computer Interaction, Okinawa, Japan, pp. 35-38, 2017.

[25] Li Liu, Shujing Lu, Yue Lu, Ching Y. Suen, "Application of PR Techniques to Mail Sorting in China. C3S2E, pp. 1-7, 2014

[26] Ying Wen, Yue Lu, Pengfei Shi, "Handwritten Bangla numeral recognition system and its application to postal automation" Pattern Recognition 40(1). pp. 99-107. 2007.

[27] Xue Gao, Lianwen Jin, "A Vision-Based Fast Chinese Postal Envelope Identification System", J. Inf. Sci. Eng. 28(1), pp. 31-49, 2012.

[28] Nabin Sharma, Abira Sengupta, Rabi Sharma, Umapada Pal, Michael Blumenstein, "Pin code detection using deep CNN for postal automation" In Proc. IVCNZ, pp. 1-6, 2017.