Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 14 (22): 3028-3031, 2014
ISSN 1812-5654 / DOI: 10.3923/jas.2014.3028.3031

Digitizing Latin Numerals

Enas Al-Rawashdeh, Dojanah Bader and Rawan Zaghloul
Department of Management of Information Systems, Amman Collage,
Al-Balqa'a Applied University, Jordan

Abstract: This study presents a precise system for translating the Latin numerals into a readable decimal format. The proposed algorithm is composed of two separate operations in its translation process. The first is to extract the features from the Latin numbers in order to recognize them. The second is to read the Latin number from the set of the recognized letters. The overall accuracy of the proposed algorithm reaches 90%. Actually, this type of applications puts high demand on the efficiency of the first operation in the translation process. Therefore, the better the recognition, the better the results achieved.

Key words: Projection, feature extraction, Latin numerals, letter recognition

INTRODUCTION

The core concept of every OCR system is to recognize the characters from an imported image (Caluori and Simon, 2013; Rehman and Saba, 2012). In other words, OCR systems are responsible for the conversion of a written text, whether handwritten or printed, into digital format (Sharma *et al*., 2013; Sumathi and Karpagavalli, 2012). In spite of the enormous researches that introduced several algorithms for character and numeral recognition in different languages such as English, Chinese, Japanese and Arabic (Sabbour and Shafait, 2013; Patil and Shimpi, 2011; Arora *et al*., 2011), poor attention was given to Latin numerals, although it can be used in many applications such as the recognition of Latin letters in clocks and watches, movies' copyright dates, sporting events and oxidation states, etc. Thereby, this study spots the light on developing a technique for recognizing Latin numerals from an imported image. Since, Latin numerals are mainly based on the following set of Roman symbols (I, V, X, L, C, D, M), therefore, recognizing this set of letters is mainly based on the techniques of character recognition as a part of the process of translating Latin numerals into decimal ones.

In spite of the huge research on the area of character recognition, there is a lack of research in Roman numerals. Thus, to the extent of our knowledge, this study is one of the minority works on the recognition of Roman numerals. Several of the existed researches worked on numeral recognition of Arabic digits, whereas Diem *et al*. (2013) conducted a comparison between a set of the Arabic digits numeral recognition techniques to evaluate their performance. In addition, Ghaleb *et al*. (2013) concentrated on the recognition of the handwritten numerals using the vertical and horizontal strokes for feature extraction.

However, Latin numerals are composed of a set of letters. Tremendous researches had been done in this field, where Pradeep *et al*. (2011) proposed a technique to recognize English characters based on a diagonal feature extraction using multilayer neural network. While, Wang and Sajjanhar (2011) produced another technique for offline handwritten characters. Their study was based on the polar coordinate classification method. They proved that their study on zoning feature extraction based on the polar classification is better than that of the Cartesian coordination, whereas in our study, we rely on the recognition technique that was proposed in one of our earlier studies (Zaghloul *et al*., 2012), where it used the recursive horizontal projections to recognize the Arabic numerals.

METHODOLOGY

The design of our approach mainly composed of two parts as depicted in Fig. 1, the first part is to recognize each letter separately, followed by the second part in which we proposed a technique to read a number from the set of the recognized letters. In the proposed technique, we assume that the range of numbers to be recognized is between (I) and (CMXCIX) in uppercase. Thus, the set of letters in this range are: {I, V, X, L, C, D, M}. Let us now consider the steps of the proposed technique:

Corresponding Author: Enas Al-Rawashdeh, Department of Management of Information Systems, Amman Collage, Al-Balqa'a Applied University, Jordan

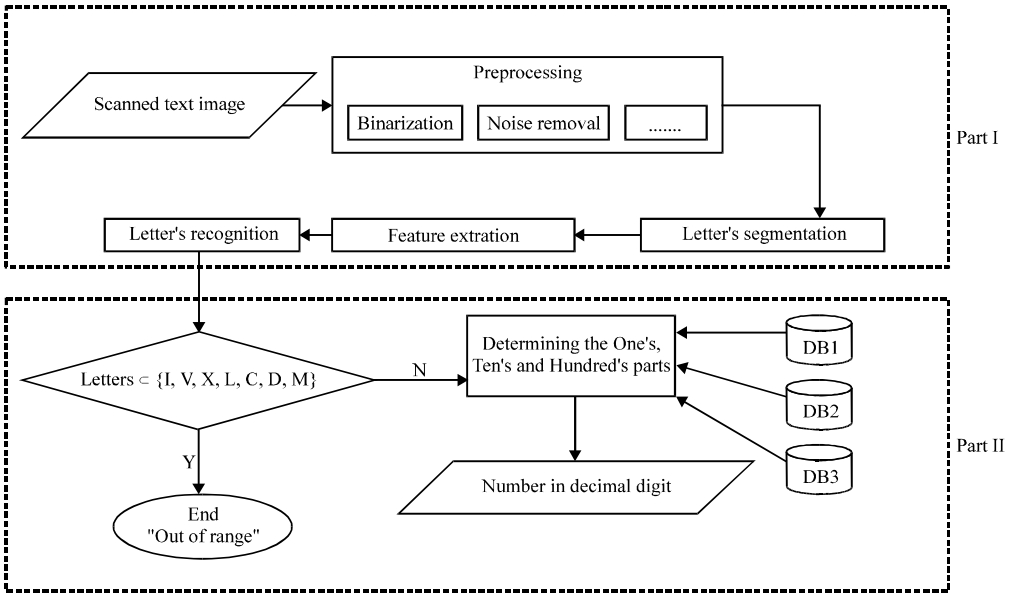


Fig. 1: Proposed model

Part I-Letters recognition: This part consists of the following steps:

- Importing the image of Latin number by a scanner or camera, etc
- Preprocessing the imported image by applying the Binarization on the image (to convert the image into black and white), followed by thinning and noise removal from the binary image
- Segmentation of the preprocessed image by breaking the image into sub images one for each letter. This step involves the identification of the boundaries of the letter and separating it for further processing
- Feature extraction for each letter separately. We employed the technique proposed by Zaghloul *et al.* (2012) where the features are extracted according to the loop detection technique and the horizontal projections of the letters as shown in Fig. 2
- Recognize the letters correctly according to the results of the projections

Part II-Reading numbers: In this part, we built three databases, the first is to represent the One's part of the number, the second is to represent the dozens (Ten's) part of the number and the third one is to represent the Hundred's part of the number as depicted in DB1, DB2 and DB3 respectively as shown in Fig. 3. This part consists of the following steps:

I		C	
V		D	
X		M	
L			

Fig. 2: Standard Latin numerals and their projections

- Compare each letter with the letters: {I, V, X, L, C, D, M}. So proceed in the following steps only if the recognized letters belong to this set
- Determine the number of letters in the image to be used separately in the next step
- Now, start by passing the letters (one after the other) from right to left. The first letter from the right must be compared with DB1, if the first letter belongs to the letters in DB1 then, move to the next letter and compare the first two letters respectively with DB1 if this pattern of two letters still belongs to DB1, we move to the next letter and compare the new pattern (of three letters) with DB1, continue in this procedure until you reach the last letter or when no match is found with DB1. If there is no match, then return one step backward and find the number that represent the previous pattern from DB1 to be its One's part as shown in Fig. 3

- After determining the One's part of the number, continue with the remaining letters. Start from the first letter in the remaining part and continue with the same procedure as in step 3 but comparing the letters with DB2 until no match is found. Now, the compared pattern at this step represent the Ten's part of the number
- After determining the Ten's part of the number, continue with the remaining letters. Start from the first letter in the remaining part and continue with the same procedure as in step 3 but comparing the letters with DB3 until no match is found. Now, the compared pattern at this step represent the Hundred's part of the number

It is important to note that at the beginning of the comparison in step 3 if the first letter does not belong to DB1 then the One's part of this number is 0. Equivalently, if the first letter in the second pattern does not belong to DB2 then the Ten's part of this number is 0, also if the first letter in the third pattern does not belong to DB3 then the Hundred's part of this number is 0.

For example, as depicted in Fig. 4 for digitizing the tern (CCXVII), start by passing over the characters from the right to left. So we start with "I" and compare it with DB1 then we move for the next character "II" and compare it with the previous one with DB1 and continue for "VII", but we stop in "XVII" because its not belong to DB1 by order. So we go to one step backwards for "VII" then comparing with DB1 and see the "VII" =7, so the One's of this number is 7.

Then we continue with the remaining characters. We have to determine this number and then we have a character "X", compare it with DB2 and see that it belonged to a DB2 then move on to the next character "CX" and compare it with the previous one with DB2 at this step, stop and go back one step backwards for "X" then comparing with DB2 and see the "X" = 1, so the Ten's of this number is 10.

DB1 (One's)		DB2 (Ten's)		DB3 (Hundred's)	
1	I	10	X	100	C
2	II	20	XX	200	CC
3	III	30	XXX	300	CCC
4	IV	40	XL	400	CD
5	V	50	L	500	D
6	VI	60	LX	600	DC
7	VII	70	LXX	700	DCC
8	VIII	80	LXXX	800	DCCC
9	IX	90	XC	900	CM

Fig. 3: Databases used in this methodology

Then we continue with the remaining characters. We have to determine this number and then we have a character "C", compare it with DB3 and see that it belonged to a DB3 then move on to the next character "CC" and compare it with the previous one with DB3. Here we come to the last character, then comparing "CC" with DB3 and see the "CC" = 2, so the Hundred's of this number is 200. So, "CCXVII" → "217".

RESULTS AND DISCUSSION

The experiments conducted over MATLAB environment. The dataset was a mixture of handwritten and printed Roman numerals scanned images. The handwritten samples were generated from different users. Figure 5, shows a sample of the examined dataset.

The results of recognizing the Latin letters, using the algorithm proposed by Zaghloul *et al.* (2012), are as depicted in Table 1. As you can see the printed letters gains higher recognition rates in comparison with the handwritten ones. This is reasonable, since, the handwritten writings suffers from ambiguity and inaccuracy and other problems such as noise, etc.,

However, after recognizing the letters, reading the combination of the resulted recognized letters by the proposed procedure will generate the digital equivalent number. In other words, if the Latin letters were recognized correctly, our algorithm guarantees the digitization of the Latin number.

Finally, it is worth to mention that applying better OCR systems will increase the recognition rate of the letters and consequently will increase the overall accuracy of the proposed algorithm.

Table 1: Recognition rates of the Latin letters (I, V, X, L, C, D, M)

	Recognition rate (%)	
Letters	Handwritten letter	Printed letter
I	88	95
V	85	90
X	86	93
L	88	95
C	90	93
D	96	99
M	85	90

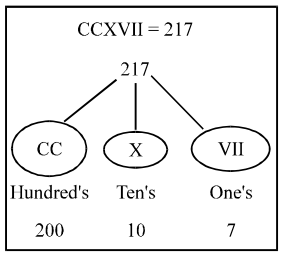


Fig. 4: Latin number with translated example

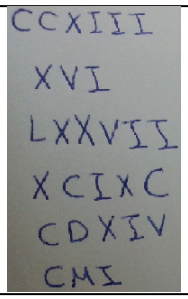
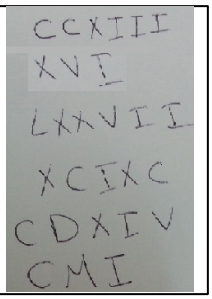
CCXVIII XVI LXXVII XCIXC CDXIV CMI	CCXVIII XVI LXXVII XCIXC CDXIV CMI		

Fig. 5: A sample of the handwritten and printed numerals

CONCLUSIONS

In summary, the overall accuracy of the proposed algorithm fully depends on the accuracy of the OCR algorithm that is adopted. In this study, we utilized from one of our recently proposed OCR (Zaghloul *et al*., 2012). It reaches at 90% according to the use of handwritten and printed samples. Thus, choosing a better OCR algorithm will improve the results.

Regarding to the performance and complexity of the number recognition (Part II of the proposed algorithm), it depends on the length of the Roman number. It achieved precise results unless there were some recognition errors generated from the results of part I (OCR). We can say that we can fully classify the numbers using the proposed algorithm if we guarantee the recognition of the Latin letters.

REFERENCES

Arora, S., D. Bhattacharjee, M. Nasipuri, D.K. Basu and M. Kundu, 2011. Complementary features combined in a MLP-based system to recognize handwritten devnagari character. J. Inform. Hiding Multimedia Signal Process., 2: 71-77.

Caluori, U. and K. Simon, 2013. An OCR concept for historic prints. Proceedings of the 10th IS&T Archiving Conference, April 2-5, 2013, Society for Imaging Science and Technology, Washington, DC., USA., pp: 143-147.

Diem, M., S. Fiel, A. Garz, M. Keglevic, F. Kleber and R. Sablatnig, 2013. ICDAR 2013 competition on handwritten digit recognition (HDRC 2013). Proceedings of the 12th International Conference on Document Analysis and Recognition, August 25-28, 2013, Washington, DC., pp: 1422-1427.

Ghaleb, M.H., L.E. George and F.G. Mohammed, 2013. Numeral handwritten Hindi/Arabic numeric recognition method. Int. J. Sci. Eng. Res., Vol. 4.

Patil, V. and S. Shimpi, 2011. Handwritten English character recognition using neural network. Elixir. Comp. Sci. Eng., 41: 5587-5591.

Pradeep, J., E. Srinivasan and S. Himavathi, 2011. Diagonal based feature extraction for handwritten alphabets recognition system using neural network. Int. J. Comput. Sci. Inform. Technol., 3: 27-38.

Rehman, A. and T. Saba, 2012. Neural networks for document image preprocessing: State of the art. Artif. Intel. Rev., 10.1007/s10462-012-9337-z

Sabbour, N. and F. Shafait, 2013. A segmentation-free approach to Arabic and Urdu OCR. Proceedings of the IS&T/SPIE Electronic Imaging, February 3-7, 2013, California, USA.

Sharma, O.P., M.K. Ghose, K.B. Shah and B.K. Thakur, 2013. Recent trends and tools for feature extraction in OCR technology. Int. J. Soft Comput. Eng., 2: 220-223.

Sumathi, C.P. and S. Karpagavalli, 2012. Techniques and methodologies for recognition of Tamil typewritten and handwritten characters: A survey. Int. J. Comput. Sci. Eng. Survey, 3: 23-35.

Wang, X. and A. Sajjanhar, 2011. Polar Transformation System for Offline Handwritten Character Recognition. In: Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, Lee, R. (Ed.). Springer, USA., ISBN: 978-3-642-22287-0, pp: 15-24.

Zaghloul, R.I., D.M.K.B. Enas and F. AlRawashdeh, 2012. Recognition of Hindi (Arabic) handwritten numerals. Am. J. Eng. Applied Sci., 5: 132-135.