Sindh Univ. Res. Jour. (Sci. Ser.) Vol. 47 (3):395-398 (2015)

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Interactive Thinning for Segmentation-based and Segmentation-free Sindhi OCR

D. N. HAKRO[++], S. A. AWAN*, M. MEMON, A. M. AAMUR, G. N. MOJAI**

Institute of Information and Communication Technology, (IICT), University of Sindh, Jamshoro

Received 2nd February 2015 and Revised 16th July 2015

Abstract: Optical Character Recognition (OCR) is converting image based images into editable text so the text written in image form is available for editing purpose. The thinning technique can be applied in preprocessing stage or after segmentation of words and characters from a text image when features are extracted to differentiate the characters. Thinning is to decrease the thickness of strokes and finding out one pixel skeleton of the character image. In this paper we present an iterative and interactive thinning algorithm for Sindhi script step by step. Our thinning algorithm removes pixels by preserving connectivity and pattern of image intact. The process can be stopped and checked with pixel based editor for the connectivity patterns. This algorithm can be used with segmentation-based Sindhi OCR and segmentation-free OCR. The algorithm along with application is tested on Sindhi line text and individual characters and the results are presented. The algorithm and the application can also be applied with other language scripts. The presented work is a part of research done on Sindhi OCR.

Keywords: Optical Character Recognition, Thinning, Feature extraction, Sindhi.

1. INTRODUCTION

Thinning is the basic step usually used in Optical Character Recognition as well as number of other image processing applications in which binary objects, shapes or patterns in an image are reduced to the size of one pixel thick (Gonzalez *et al.,* 2005). For the optical character recognition the feature extraction is normally depending upon the distinctive features. To process these features as well as processing of the images need less and less information to be processed so there is a situation when thinning is needed. This is because the thinned patterns are tending to occupy less memory to the original ones. Thinning algorithms applied on hardware or software can produce different levels of distortion (Zhang and Suen,1984). Thinning as well as skeletonisationis used in different literature. According to Lam *et al.* (1992)recognition was performed on thinned characters by Sherman (1959),Duetsch (1968)and Alcorn (1969). After that some of the circuit boards were designed based on this purpose.

Louisa Lam *et al.* (1992) reported a wide variety of the application of the thinning algorithms such as white blood cells analysis, quantitative matallography,and visual analysis of industrial parts circuit boards and other together with the necessary part for the optical character recognition (Lam *et al.,* 1992). Cowell and Hussain (2001)reports that a simple search can produce 150 research papers on thinning but today the number is even higher than reported.Jagna and Kamakshiprasad (2010) proposed parallel algorithm for the thinning patterns by using 3X3 mask for the deletion of the pixels and the connectivity preservation is also discussed Zhang and Suen, (1984) proposed a fast algorithm with two sub-iterations for the deletion of south-east boundary and north-west points. Shang and Yi (2007) experimented with the binary images by using two pulse coupled Neural networks and claimed that in future the applications of their algorithm will be studied. They used filling technique to fill the region for the obtaining of the inner and outer image. The firing step obtains the thinned image which is decided and the final thinned image or the process is repeated.(Zhou *et al.* (1995) presented a novel thinning algorithm based on single pass. Bitmap and flag map are used concurrently for the decision of pixel to be deleted. Problems in existing algorithms and their solutions are presented and a smoothing template is also proposed for the smoothing of thinned image.Chiu and Tseng (1997) presents a handwritten Chinese character thinning algorithm with feature preservation. The thinning algorithm contains the phases of direction codes determination in which direction of the pixels are decided. The second phase contains the block division in fork segments and strokes. The third phase is the extraction of the skeleton segments while removing noise and hair branches. The final step is the connection of those skeleton segments extracted in third step. Holt *et al.* (1987) proposed a parallel thinning algorithm which requires calculation at each iteration step.

[++]Corresponding author: Dil Nawaz, email: dill.nawaz@gmail.com
*Benazir Bhutto Shaheed University, Cheel Chowk, Lyari Karachi, Sindh, Pakistan.
**Institute of Mathematics & Computer, Sindh University, Jamshoro, Pakistan

Comparison with Cowell and Hussain (2001):

This work is inspired from Cowell and Hussain (2001) but differs in many aspects. Cowell and Hussain (2001) presented their work on isolated Arabic characters whereas the current study is equally useful for the isolated characters as well as the compound characters which is a clear advantage because the created software can be used for Sindhi characters, words and the full text lines. It is clearly seen in Cowell and Hussain (2001) that the isolated characters are in red color which means the thinning is performed on color images whereas our approach is purely based on binary images resulting less computation and clear results and the images can be edited easily.

Interactive Thinning:

An application was created in MATLAB to skeletonize the text. The text can be from other scripts. Sindhi is the largest extension of Arabic script, hence other scripts adopting Arabic script can also be used to skeletonize images of these scripts. For thinning of the images of the Sindhi text different routines are created to thin the image containing a character, word or a full sentence. These routines are called when they are needed and embedded in a single interface of the MATLAB. The text image is loaded into one of the windows. The text image is shown at every stage and that can be edited for the fine tuning as shown in **(Fig.1).** The image is further called in another pixel based editor where the image can be edited up to pixel based where a desired pixel is removed or added if needed. The image can be started and stopped at any iteration so that the connections of the character can be checked and according to situation the pixels may be added or removed. The iterations can be controlled according to the character image which means more thick character will need more iteration and finally at any given iteration final result is automatically saved to the disk.

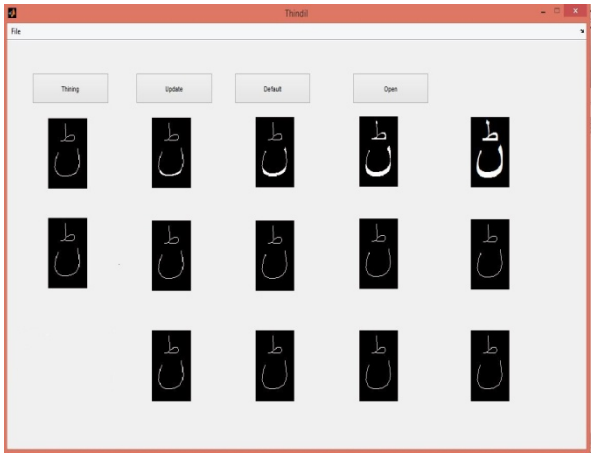


Fig. 1: Thinning steps of character bay "ٻ"

Thinning in OCR:

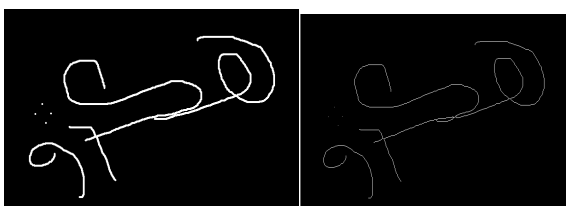
Thinning approach can be used in preprocessing and it is also needed in OCR before the features are extracted. The thinning algorithm is applied in both types of OCR such as segmentation-based and segmentation-free OCR. Thinning algorithm is crucial and helps to simplify the object recognition and analysis, pattern recognition and analysis and feature extraction (Shang and Yi, (2007). A thinned image is more appropriate as compared to the original one because it contains the only necessary part or skeleton of the character image. It was widely used and documented during high quality Latin recognition(Cowell and Hussain, 2001). Thinning also saves the amount of space and reduces the image data which will be processed in the next stage. The other benefit is that after thinning the shape of character can easily be analyzed.

Thinning algorithm:

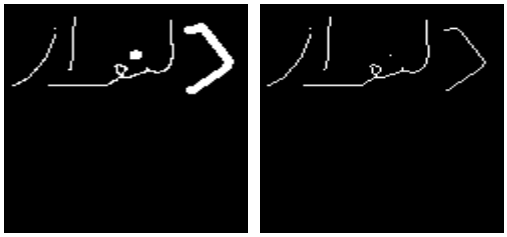
The maximum iterations can be set via passing a value and can be set to stop the iteration process. An array of zeros 2 pixel wider than original image is created for masking and the images is scanned for white pixels. Then the pixels are checked for touching white pixels with the black pixels which form the background. Three conditions for checking the neighboring pixels are checked and the last pixel that touches the background is the candidate pixel for removing so that a character may be skeletonized. The algorithm is using 3x3 window and edge information for deletion or survival of the pixel. The process is repeated on partially thinned image to ensure the image contains one pixel wide strokes of the character and no more pixels are removed.

2. RESULTS AND DISCUSSION

The creation of application using thinning algorithm is a part of research done on Sindhi OCR and multi-script OCR. The algorithm was applied on Sindhi text images created with custom built software and handwritten images. The images were in the form of isolated characters, ligatures, words and a sentence of Sindhi script. Text images of Sindhi language were thinned in created application and some of the results are presented. **(Fig.2 (a and c)** illustrates the original images of author's nameand cast and **(Fig. 2 (b and d)** shows the output images after thinning.



(a) (b)



(c) (d)

Fig.2: Thinning of an image: (a and c) original image (b and d) image after thinning

The algorithm was applied text images created with custom built applicationHakro *et al.* (2015) and handwritten images of Sindhi language. The images created with custom built application were in various fonts, styles and sizes. These text images of Sindhi script were thinned, used to extract features and recognized in Sindhi OCR. The algorithm performs well and the characters are skeletonized and used to extract features as shown in **(Fig. 3).**

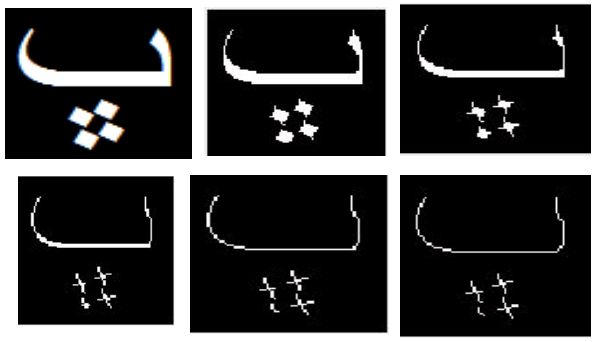
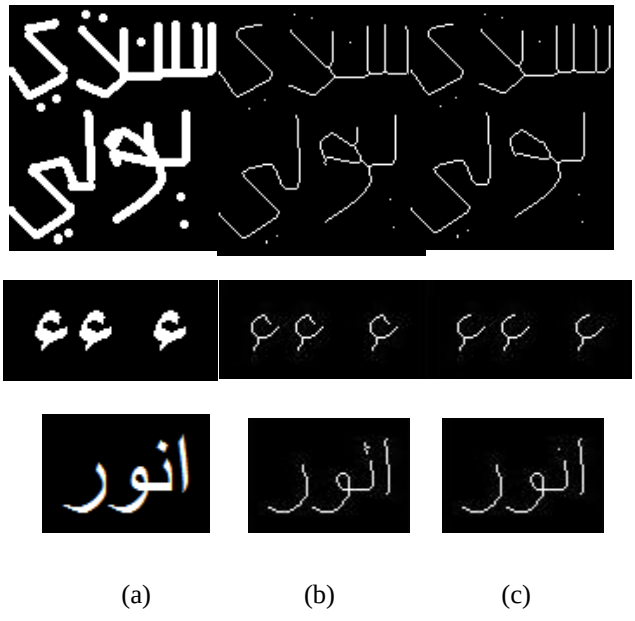


Fig. 3: Thinning of an image step by step (Left to right)

The algorithm results in some of the hairy tails. These hairy tails are removed by stopping the iterative thinning process and opening resultant image in pixel editor and cleaning undesired pixel values as shown in **(Fig. 4(a),** theimage with hairy tails in isolated character "**پ** " and **Fig. 4(b**) is the image after removing hairy tails. **Fig. 5 to Fig.7** show the thinning and removing tails. Figure 5(a) illustrates original image of Sindhi words and characters (b) shows the images after thinning with some of remaining tails and **Fig. 5 (c)** shows the final images after thinning algorithm.

(a) (b)

Fig.4: (a) Original image(b) Removing of undesired pixels



(a) (b) (c)

Fig. 5: (a) Thinning (b) removing of undesired pixels (c) result

Fig. 6(a) illustrates the original text image of Sindhi characters in a single image and Figure 6(b) shows the images after thinning process. The algorithm works for multiple images simultaneously so it can be used with any text image containing Sindhi words and characters and it is useful to use for segmentation free OCR for Sindhi script. Fig. 7 (a) shows the isolated character images and Fig. 7 (b) results after thinning.

(a)

(b)

Fig. 6: Thinning of Sindhi text line:

(a) Original text line image (b) After thinning

(a) (b) (c)

Fig.7: Thinning of single character of Sindhi script (a) Original (b) After thinning

3. CONCLUSION

Sindhi script is an extended version of Arabic script and contains all problems of Arabic script in addition of its own script peculiarities.The algorithm was tested on various fonts and various scripts and the results are beyond the scope of this paper as this work is part of Sindhi OCR development in which a huge database of Sindhi and other scripts was created and images were thinned for this purpose. The application helped in thinning of words and characters used in Sindhi OCR.

4. FUTURE WORK

The software is working with machine printed and hand written characters and words for Sindhi script which is equally useful for both segmentation-free and segmentation-based character recognition systems. As this is a part of Sindhi and Multi-script OCR research in this regard, some basic experiments were performed for other scripts and the complete system performing thinning of other scripts will be presented in future.

REFERENCES:

Alcorn T. M. and C. W. Hoggar,(1969). "Pre-processing of data for character recognition," Marconi Rev., vol. 32, 61-81,

Chiu, H. P. and D.C. Tseng, (1997).“A feature-preserved thinning algorithm for handwritten Chinese characters”, Signal Processing, Vol.: 58(2), 203–214,

Cowell J. and H. Fiaz (1992). “Thinning Arabic character feature extraction“, IEEE Transactions on Pattern Analysis Machine Intelligence, Vol. 14, No.11, 869-885,

Deutsch, E. S. (1968) "Preprocessing for character recognition," in Proc. IEEE NPL ConfPatt. Recogn. (Teddington), 1, 179-190.

Holt. C. M., A. Stewart. M. Clint. and R. H. Perrott. (1987).“An improved parallel thinning algorithm”, Commun. ACM, Vol.: 30(2), 156 - 160,

Hakro, D. N., Z. Talib. G. N. Mojai. (2015), ‘Multilingual Text Image Database for OCR ', *Sindh University Research Journal (Science Series)* 47(1), 181-186.

Jagna A. and V. Kamakshiprasad (2010) “New Parallel Binary Image Thinning Algorithm”, Asian Research Publishing Network (ARPN). Journal of Engineering and Applied Sciences, Vol. 5, No. 4,.64-67. ISSN: 1819-6608.

Lam L., S. W. Lee, S. Y. Suen, (1992). “Thinning methodologies - A comprehensive survey, “IEEE Trans. Pattern Analysis and Machine Intelligence, 869-885.

Rafael C. Gonzalez, R., E. Woods and S. L. Eddins, 2005 “Digital Image Processing using MATLAB”, Pearson Education, Third Indian Reprint, 370- 375.

Sherman, H. (1959) "A quasitopological method for the recognition of line patterns." in Proc. Int. Conf. on Inform. Processing (Paris, France), 232-238.

Shang, L. and Z.Yi, (2007).“A class of binary images thinning using two PCNNs”, Neurocomputing, Vol.: 70, 1096-1101,

Zhang T. Y. and C. Y. Suen, (1984). “A fast Parallel Algorithms for Thinning Digital Patterns”, Research Contributions, Communications of the ACM. 27 (3): 236-239,

Zhou, R. W. C.Quek and G. S. Ng, (1995) “A novel single-pass thinning algorithm and an effective set of performance criteria”, Pattern Recognition letters, Vol. 16, 1267 – 1275,