Heterogeneous Binary Image Thinning Algorithm

Amgoth Jagan
Department of Computer Science and Engineering
B V Raju Institute of Technology
Vishnupur, Narsapur, Medak (Dist) Telangana

ABSTRACT

Image thinning plays an important role in image analysis and pattern recognition applications. As a pre-processing step, image thinning algorithms are used to reduce the amount of data to be processed while preserving the relevant details. The proposed heterogeneous image thinning algorithm is aimed at thinning a variety of binary image patterns to achieve the desired result. This algorithm is based on conditions that are met by a set of rules or templates. These rules are applied simultaneously to every pixel in the image. To check the suitability of two fundamental requirements of thinning, namely the speed of thinning and the quality of the skeletons. The thinned images preserve geometrical and topological properties as well as the one-pixel-wide central medial axis.

Keywords: heterogeneous, medial axis, geometrical, topological, and template.

1. Introduction

The thinning of images plays an important role in image pre-processing. It is an instrumental preprocessing step in many applications, such as document analysis, image compression, data compression, fingerprint classification, and pattern recognition. The thinning algorithm converts an input binary image into one-pixel skeletons with nearly thin lines, curves, and arcs. When applying the thinning algorithm to binary images, the thinned image must preserve relevant information such that it will be useful for later processing, such as extracting features from a pattern.

There are two types of thinning algorithms, such as iterative and non-iterative, based on the approaches used for thinning [Lam, Lee, and Suen, 1992]. With a non-iterative thinning algorithm, a one-pixel thick pattern is obtained in the first pass. Iterative algorithms, on the other hand, delete unwanted pixels while preserving the boundary pixels step by step until the pattern is one pixel thick. In iterative algorithms, the boundary pixels of an image pattern are processed one by one and step by step. Image pixels are processed simultaneously, layer by layer, in parallel algorithms. The memory consumption of sequential algorithms is much lower than that of parallel algorithms. Parallel image thinning algorithms process all the border pixels at once and produce a one-pixel thick pattern.

A successful thinning algorithm should produce thinned patterns with the following properties: 1. The thinned pattern should be a single-pixel width. 2. There will not be much deviation from the central pixel. 3. The thinned pattern should maintain topological and geometric properties. 4. The algorithm must take less memory and time, and most importantly, the skeletons produced must be in such a way that it is possible to reconstruct the original image.

All algorithms may not hold all the characteristics, but a successful algorithm holds most of them.

2. The principle followed in the proposed iterative-based thinning algorithm

In an iterative-based image thinning algorithm, the image is thinned simultaneously in two iterations (iteration 1 and iteration 2). To retain the topological properties of the thinned image, we need to preserve endpoints and connected points whose BP value is 2. Iteration 1 deletes image pixels with BP values ranging from 3 to 6 and AP values of 1. Border pixels are defined as two-pixel-wide horizontal or vertical patterns with BP values of 3 or 5 and AP values of 1. If we apply the same condition to the horizontal line, which has a two-pixel thickness, we will erase the entire line. Since there is no clarity in processing a two-pixel thick image to one-pixel thick, because, the desired output may be either the top row or the bottom row, and the left column or the right column. So, the vertical line should be preserved at two pixels thick in iteration 1.

The horizontal and vertical lines are thinned by considering the four-connected neighbors of pixels. In horizontal two-pixel thick pattern images, we preserve pixels belonging to the bottom row, and in the vertical right column. The conditions to be satisfied to delete pixels are P2 and P8 = 0, and BP value in the range of 3 to 6 & AP value is 1. These conditions remove pixels both in horizontal and vertical patterns. For vertical lines, the pixel to be removed is the top pixel belonging to the leftmost column. If it is a horizontal line, the pixel to be deleted is the left pixel belonging to the top row.

Iteration 2 thins out these cases as well as the two pixel-wide diagonal lines. There are two classes of pixels in two pixel-wide diagonal lines. A group of pixels with a BP value of 2 and an AP value of 1, and another group with a BP value of 3 and an AP value of 2. Thus, we don’t include the AP value in the pixel deletion condition in iteration 2. Based on these four neighbor pixels, the conditions are derived. In diagonal lines, we preserve either the leftmost pixel in each of the two pixels belonging to a row or the bottom pixel in each of the two pixels belonging to a column.

Except for these two pixel-wide image patterns, all remaining patterns are processed in the first iteration. Thus, the processing of two pixel-wide patterns must be done separately. It is observed that all two pixel-wide patterns and non-zero neighbors are border pixels. As a result, when dealing with two-pixel wide lines, we take this characteristic into account. Pixels that do not meet the median axis condition, and have a BP value in the range of 3 to 6 and an AP value is 1 are marked for deletion. The marked pixels are checked again for connectedness and are preserved even though they are marked for deletion.

The following are the conditions that were followed in Iteration 1 and Iteration 2.

Iteration-1

1. The BP value is in the range of 3 to 6
2. AP value is 1
3. $P_2 = 0$ Л $P_8 = 0$

Iteration-2

1. The BP value is in the range of 2 to 6
2. a) $P_2 = 0$ Л $P_4 = 0$ **or**

   b) $P_3 = 1$ Л $P_4 = 1$ Л $P_8 = 0$ **or**

   c) $P_6 = 1$ Л $P_7 = 1$ Л $P_2 = 0$

3. Result analysis

The skeleton points of the image capital alphabets superimposed on the input image of size 557x446 as shown in figure 1. After thinning the input image, the skeletal points obtained are 3262 against the 41910 object points, obtaining the thinning ratio of 92.22%. The thickness of the obtained skeleton is well connected and the resultant pixels are 0.999995, almost equal to 1 (indicating one pixel thick).

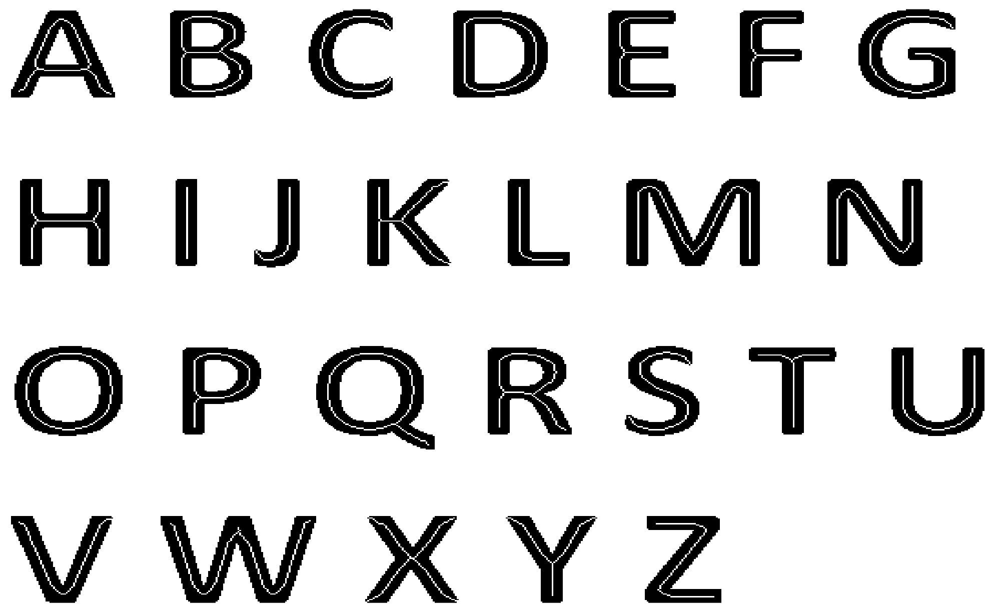

Figure.1 Capital English alphabet (superimposed)

The input image of size 531x130 is shown in the figure. 2(a) and the thinned image shown in figure 2(b). The thinning ratio obtained from the input image is 58.665. The resultant skeleton was well connected and produced in 124ms with 0.99993 thickness.

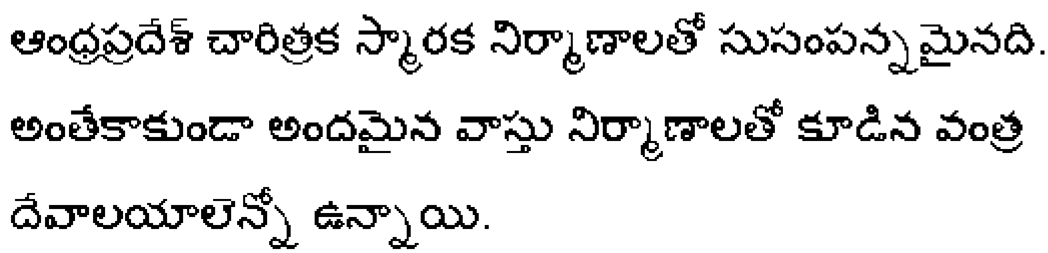

Figure. 2a Printed Telugu Text Input Image

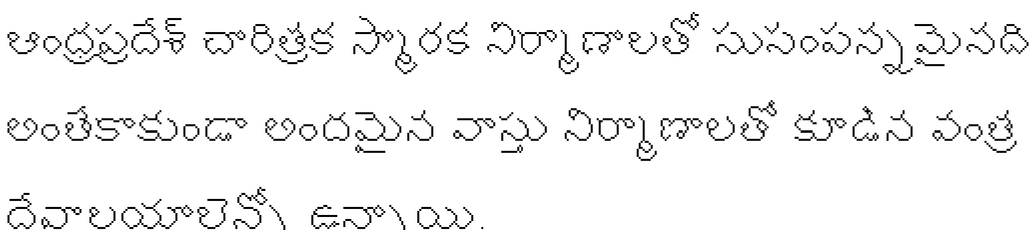

Figure. 2b Printed Telugu Text Thinned Image

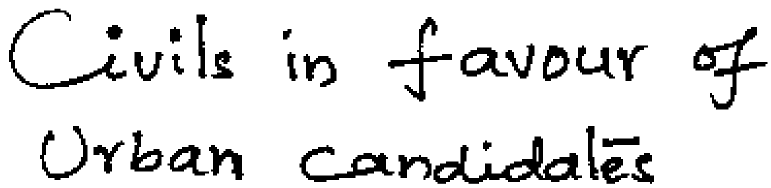

Figure 3a. Hand Written English Text Input Image

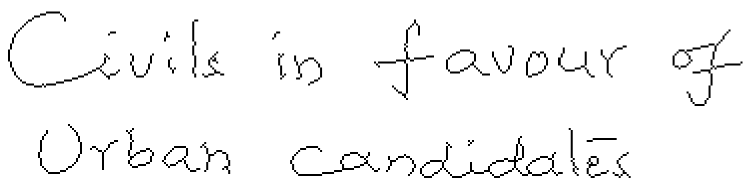

Figure 3b. Hand Written English Text Thinned Image

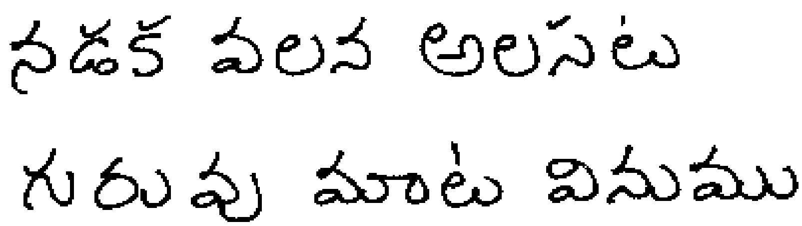

Figure 4a. Hand Written Telugu Text Input Image

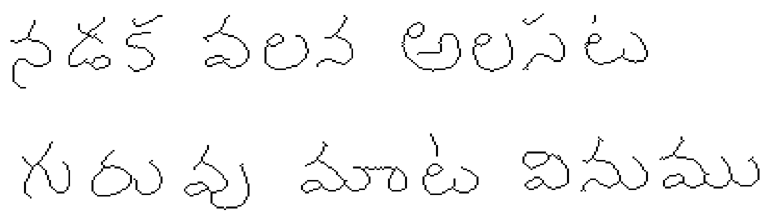

Figure 3b. Hand Written Telugu Text Thinned Image

The proposed heterogeneous image thinning algorithm was tested on various image patterns and produced a good thinning ratio in minimum execution time. All thinned images are produced one-pixel-thick with well-connected points.

4. Conclusion or Summary

The proposed heterogeneous image thinning algorithm has been tested against different image patterns like printed and handwritten characters of English, Telugu, Hindi, Tamil, and Malayalam with different sizes and orientations. For all types of images, the proposed algorithm producing good skeleton images with one-pixel-thick in minimum amount of time by maintaining topological and geometric properties.

References

1. "Gabor Nemeth and Kalman Palagyi", Parallel Thinning Algorithm Based on Ronse's Sufficient Conditions for Topology Preservation, Research Publishing Service, pp.no.1-12, Aug.16. 2009

2. "Saad Harous and Ashraf Elnagar", Handwritten Character-Based Parallel Thinning Algorithm: A Comparative Study. Univ. of Sharjah Journal of pure & Applied Sciences, Feb, Vol.6, no.1. pp.no.81-100. 2009.

3. "Gulshan Goyal, Maitreyee Dutta and Er. Akshay Sridhar", A Parallel Thinning Algorithm for Numerical Pattern Images in BMP Format, published in International Journal of Advanced Engineering & Application, Jan, pp.no.197-202. 2010.

4. "G.S. Ng, R.W. Zhou and C. Quek", A Noval Single Pass Thinning Algorithm IEEE Transaction on System Man and Cybernetics, 8 Sep. 1994.

5. "Peter Kardos, Gabor Nemeth, and Kalman Palagyi", An order independent sequential thinning algorithm", IWCIA, LNCS 5852, pp. 162-175, Springer – Verlag Berlin Heidelberg 2009.

6. "Maher Ahmed and Rabab Ward," A Rotation Invariant Rule-Based Thinning Algorithm for Character Recognition", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.24, pp.no.1672-1678, No.12, Dec-2002.