Design and Implementation of Algorithms for Extraction and Representation of Morphological Features

Kapila Sharma
Department Of CA, SMIT, Rangpo, Sikkim, INDIA

Mohan P. Pradhan
Department Of CSE, SMIT, Rangpo, Sikkim, INDIA

ABSTRACT
Automated information extracted from topographic sheet plays an important role in assisting researchers to extract identifiable features in the field of remote sensing and geographical information system. Extraction and representation of morphological patterns pertaining to an aspect of interest plays a pivotal role while making a quality assessment of the same. The confidence of the assessment procedures heavily relies on the effectiveness with which the patterns are extracted and represented. These extracted features can be used for performing various inferential analyses that aids in making qualitative as well as quantitative assessment of the features. This work aims at extracting various morphological features from topographic sheet automatically which further reduces time and effort requirement compared to that of traditional manual system.

Keywords
Topographic sheet, Extraction.

1. INTRODUCTION
Extraction and identification of morphological features into different identifiable categories is an important aspect of feature processing. In studies related to biometrics, minutiae play a pivotal role in making quality decisions related to various aspect of interest such as finger print, iris and many more. Minutiae in biometrics are specific points in a finger image. Minutiae are major components of morphological features. Minutiae include: Ridge ending- the abrupt end of a ridge, Ridge bifurcation- a single ridge that divides into two ridges, Short ridge, or independent ridge- a ridge that commences, travels a short distance and then ends, Island - a single small ridge inside a short ridge or ridge ending that is not connected to all other ridges, Ridge enclosure - a single ridge that bifurcates and reunites shortly afterward to continue as a single ridge, Spur - a bifurcation with a short ridge branching off a longer ridge. Crossover or bridge-a short ridge that runs between two parallel ridges, Delta - a Y-shaped ridge meeting, Core - a U-turn in the ridge pattern. For the identified features attribute details are to be associated which can be used for the purpose of making statistical inferences regarding the features. A repository for the same is to be created and its population is to be done.

2. RELATED WORK
Table 1. Related Work

<table>
<thead>
<tr>
<th>Ref No</th>
<th>Name of Paper</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1]</td>
<td>Contour line recognition from scanned topographic maps</td>
<td>Presents a method for contour line recognition starting from scanned topographic map. This paper uses local geometric properties to recognise contour lines based on</td>
</tr>
</tbody>
</table>

|   | Recognition of road and river patterns by relational matching | Automatic exterior orientatation of images which uses relational matching method. The structural description of the images are obtained by thresholding selected channels of colour images. |
| [2] | Extraction of stream channels in high-resolution digital terrain images using morphology | An approach for the extraction of stream channels from airborne laser swath mapping(ALSM)data. Applies morphological operations on an ALSM DEM to detect stream channels. The proposed method uses small-scale morphology to provide complementary results for streamline locations over a small catchment area. |
| [3] | An efficient connectivity-number-based edge detection method for binary images | A novel binary image edge detection method is presented. It directly clears the inner pixel of the images to obtain the edges. Connectivity number were sought as detectors, those pixels that have zero connectivity-number were eliminated as inner pixels from the image. |
| [4] | Automatic extraction of contour lines from topographic maps | An algorithm to extract contour lines from coloured images of scanned topographic maps. First segment the colour images using adaptive thresholding to extract basic contour structure. Noise in the image is removed using morphological operations. |

3. METHODOLOGY
the approach applied for extraction and representation of morphological features various parts of the image are represented with zeros and ones in matrix representation which define the shape of the image. The matrix is scanned in all possible locations traversing all pixels. The structural information of the image is justified with the pixel set to one (foreground) and the pixel set to zero (background) do not have any relevance to the feature of interest.
Algorithm Description: traverse a matrix with 0’s and 1’s
and with every 1 encountered check for presence of
features enclosures, lines, points.

Variable description:
Input : mat [size][size]
Output : mat [size][size]
Processing: i j
Pre : mat [size][size ] is a matrix of size 15 by 26
Post : enclosures, lines, points detected and coordinates
displayed
Step 0: start
Step 1: call check closure ()
Step2: call check line ()
Step3: check point ()
Step 4: stop

Algorithm _check closure ()
Algo_description: Method that checks if the given point in the
matrix is a part of and to closure identifies the closure.

Variable Description:
Input : mat [size][size]
Output : mat [size][size]
Processing: i j
Pre : mat [size][size ] is a matrix of with 0’s and 1’s
Post : mat [i][j] with the closures detected and
cordinates displayed
Step 0: start
Step 1: [read] i, j, row no. col no.
Step2: call visit points [size][size]
2.1 [test condition] if neighbor [0]==begin x and neighbor
[1]==begin y and visited length >2
2.1.closure [no closure].pos
2.2 [switch case] check for positions (x,y+1), (x+1,y+1),
(x+1,y), (x+1,y-1), (x,y-1), (x-1,y-1), (x-1,y),(x-1,y+1)
Step 3: stop

Algorithm _checkline ()
Algo_description: method to check if the given point in the
matrix forms lines

Variable description:
Input : mat [size][size]
Output : mat [size][size]
Processing : i j
Pre : mat [size][size ] is a matrix of 0’s and ‘s
Post : number of lines formed by each non zero
element
Step 0: start
Step 1: [initialize] temp=0
Step2: [initialize] temp_1=0
2.1 [test condition] if mat[x][y].value==1 and mat[x][y].is
used==0
2.1.1 [test condition] if mat[x][y+1].value==1 and
mat[x][y+1].is used==0
2.1.2 [test condition] if mat[x+1][y].value==1 and
mat[x+1][y].is used==0
2.1.3[test condition] if mat[x+1][y+1].value==1 and
mat[x+1][y+1].is used==0
2.1.4[test condition] if mat[x+1][y+1].value==1 and
mat[x+1][y+1].is used==0
2.1.5[set] mat[x][y].is used=1
Step3: stop

Algorithm _checkpoint ()
Algo_description: method to check if the point in the matrix
forms points

Variable Description:
Input : mat [size][size]
4. RESULT AND DISCUSSION

Table 2. Result and Discussion

<table>
<thead>
<tr>
<th>Input image</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Input Image" /></td>
<td><img src="image2.png" alt="Output Image" /></td>
</tr>
</tbody>
</table>

| The following are the points in the matrix | 0.2 0.6 0.14 0.18 0.22 0.24 1.3 2.4 3.5 14.16 |
| The following are the lines in the matrix | Line 0: 8.1 9.1 10.1 11.1 12.1 |
| | Line 1: 8.21 8.22 |
| | Line 2: 13.15 13.16 |
| | Line 3: 14.4 14.5 14.6 14.7 |
| The following are the closure | Closure 0: 5.8 5.9 6.9 7.9 8.9 9.8 8.7 7.7 6.7 |
| | Closure 1: 12.22 13.23 14.22 13.21 |
| The following are the points in the matrix | 1.11 6.3 6.15 6.21 9.0 9.8 10.3 14.11 |
| The following are the lines in the matrix | Line 0: 0.8 1.7 2.6 |
| | Line 1: 0.10 1.11 2.12 |
| | Line 2: 0.19 1.19 2.19 3.19 4.19 |
| | Line 3: 2.0 3.0 4.0 5.0 6.0 7.0 |
| | Line 4: 3.6 4.6 5.6 6.6 7.6 |
| | Line 5: 3.12 4.12 5.12 6.12 7.12 |
| | Line 6: 5.20 6.21 7.22 8.23 |
| | Line 7: 8.7 9.8 10.9 |
| | Line 8: 8.11 9.10 |
| | Line 9: 10.12 11.11 12.10 13.9 |
| | Line 10: 12.23 13.23 |
| | Line 11: 13.1 13.2 13.3 13.4 |
| The following are the closure | Closure 0: 2.9 3.10 4.10 5.10 6.10 7.9 6.8 5.8 4.8 3.8 |
| | Closure 1: 10.18 11.19 12.18 11.17 |

5. CONCLUSION

The algorithms extraction and identification of morphological features were successfully implemented.

The algorithm implemented for identifying morphological features were capable of identifying feature like enclosure, ridge and points. This implementation can be further extended to bifurcation.

Identification of morphological features: To identify the morphological features like the enclosure, lines, points. The user is required to input the binary matrix (input image) from which one wishes to recognize the various features which were successfully identified and stored with coordinate points.
6. REFERENCES


