Aastha Chaudhary
CECS 553 – Machine Vision

Homework Assignment – 4

Contour Tracing Algorithms

Abstract

This chapter highlights the concept of tessellations and the various contour tracing algorithms which have been put into use so as to identify the contour of the moving objects. [1] There are various kinds tessellations which have been used so far to provide a basis for the tracing algorithms to proceed with the tracing of the digital images. The starting and stopping criterions of the contour tracing algorithms are identified and which criteria is better than other criterions to get a better contour of the pixels in the image. [2]

Introduction

The tessellation of a plane refers to the segmentation of a plane into multiple regions. We usually identify the planar tessellation as a combined set of closed sets T = {T1, T2, T3… Tn} which accounts for the planes which don't have any overlapping or gaps. The two terms associated here with the plane configuration of the digital image are overlapping and gaps. The overlapping occurs when certain elements of the image overlap with the other part of the image making it difficult to identify the contour of the image. While gaps here refer to the space in between the various elements of the image, say if the image is not continuous or we are dealing with the contour tracing for the alphabets then in that case we will encounter a few spaces every now and then. The contour tracing algorithms have few limitations making then unable to identify the contour properly for the 8-connected images i.e. the larger images. The various tessellations which were being identified are listed as follows- [1]

i. Triangular Tessellation
ii. Hexagonal Tessellation
iii. Square Tessellation

The pictorial representation of all the three tessellations can be referred from the images below.

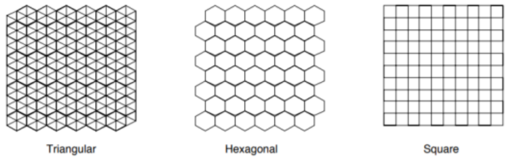

Fig. 1 Illustrating the three types of regular tessellations. [1]

Connectivity

In a Tessellation T, each of the pixels present in the grid seems to be connected to the each other in the tessellation. Each type of tessellation can be defined as a set of connected components of black pixels. Two pixels connected to each other via a common edge are called neighbours. We say that the pixels in P are connected to each other if they belong to the sequence of pixels p1, p2, … pi which belong to the pixel set P. The pixels in the sequence are called neighbours if they are present adjacent to each other in the grid. If the a black pixel has 4 adjacent neighbours then the grid is called a 4- connected grid and if there are 8 pixels near by then it is called a 8-connected grid. [2]

Contour Tracing

Contour Tracing is a method to identify a border pixel for a black pixel and then find the corresponding boundary of the entire

object/image. There are various contour tracing algorithms which are used for the detection of the border of the image. The most widely used among them are square tracing algorithm and the Moore tracing algorithm. However, these algorithms have certain limitations. They don't work as expected in case of larger images where we have more no. of connected neighbours say with 8-connected neighbours. These algorithms operated on a square tessellation which is used most widely as a tessellation for contouring algorithms.
One of the concern with the square tracing algorithm is that it is unable to identify the gaps like for example while identifying the contour of the alphabets the square tracing algorithm will not be able to find the contours by ignoring the **gaps** i.e. the **holes** present in the image. As a result, we will have to run the **hole searching algorithm** prior to running the square tracing algorithm. [2]

Square Tracing Algorithm

The square tracing algorithm works well on a 4-conneted grid of pixels surrounding a black pixel where we need to identify the border. To start with we have a set of black pixels to which we need to trace the border, we mark this pixel as the start pixel. Then we move from bottom to topmost pixel from the leftmost pixel and then to the right so as to identify another black pixel. So, every time we identify a black pixel we need to move left and every time we find a white pixel we need to right until we reach the start pixel again. [2]

The algorithm is as follows-

i. A square tessellation T having a connected component of black pixel P is the **Input**.
ii. A sequence B (b1, b2… bn) of the boundary pixels as **Output**.
iii. Begin
(find a pixel s in P to initialize B)

- Set B to empty.
- Start scanning upwards from the leftmost pixel to topmost pixel and then to the right. Once another black pixel is identified (insert it into B) and turn left otherwise turn right and keep scanning until we get a black pixel or the start pixel s.

While is P is not equal to S and the current insert is a black pixel then

- Turn towards the adjacent right pixel.
- Update the new value for P.

Else

- Turn towards the right adjacent pixel.

End while

End [1]

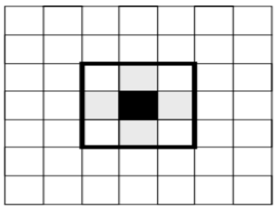

Fig. 2 Illustrating a square tessellation with its 8 connected neighbours. [1]

The square tracing algorithm fails when experimented with a large image. As a result, there was a suggestion to change the stopping point of the algorithm. For now, the stopping point of the algorithm is the second occurrence of the start pixel from where we started tracing the boundary of the figure.

The Stopping criterion suggest that the start pixel can be traversed n times so as to identify all the patterns which are present as a part of the image and were not traced or identified in the first round of the algorithm. In other words, the algorithm can be traversed multiple number of times so as to identify all the patterns which are present in the image. Or else we can stop on the second occurrence of the start pixel. [2]

This was short description of square tracing algorithm, now coming to Moore's algorithm which seems to work better on 8-connected grids.

Moore's Tracing Algorithm

The Moore's tracing algorithm works well on 8-connected grid of the pixels. It starts evaluating the pixels in the clockwise direction from the start pixel until another black pixel is identified.

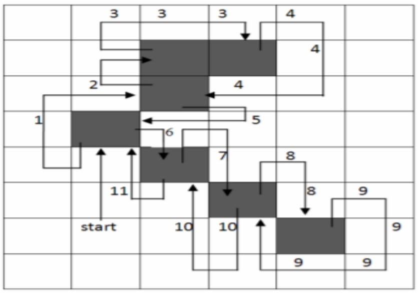

Fig. 3 Illustrating Moore's algorithm. [3]

The Moore's algorithm also has two options according to the stopping point criterion, the algorithm either stops on the second occurrence of the start point, however in that case the boundary for the large image will not be traced properly. So, we can also traverse the start point n times in order to extract or trace the boundary of all the patterns present in the image. [3]

Modified Moore's Algorithm

The Moore's algorithm was modified a bit so as to increase the system efficiency. The new algorithm requires the traversal of the image in the anticlockwise direction if no new black pixels were found. The modified Moore's algorithm is as stated below-

i. A square tessellation T containing P as set of black pixels as **Input**.

ii. A set B {b1. b2….bn} as boundary pixel values as **Output**.

iii. Begin

- Assign B to empty
- Find a black pixel and pick it as a starting point s.
- Traverse the grid from bottom to top and left to right in clockwise direction until another black pixel is identified. Let c be the next pixel to be taken into account.
- While c is in B
    - If c is black
    - Insert c in B
    - p=c
    - end if
    - Advanced to next pixel in clockwise direction
- While is c is not in B
    - If c is black
    - Insert c in B
    - p=c
    - end if
    - Advanced to next pixel in anticlockwise direction
- End while

End [3]

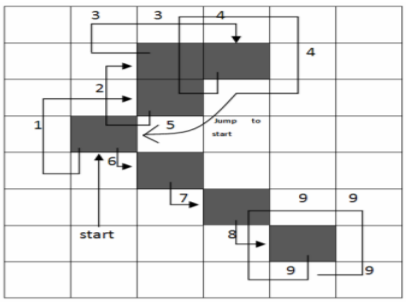

Fig. 4 Modified Moore's algorithm. [3]

Conclusion

The modified Moore's algorithm works better in comparison to the Moore's algorithm as it greatly reduces the efficiency as the Moore's algorithm depends on the stopping point criteria to improve the traversing which may increasing the work done by the algorithm. The hole searching algorithm is also not required in this which further reduces the time complexity. Henceforth, the modified Moore's algorithm proves to be efficient and better in comparison to the other two algorithms. [2]

References

1. Chapter 2, GRIDS, CONNECTIVITY AND CONTOUR TRACING Godfried Toussaint
2. P.Rajashekar Reddy, V.Amarnadh, Mekala Bhaskar "Evaluation of Stopping Criterion in Contour Tracing Algorithms " , International Journal of Computer Science and Information Technologies, Vol. 3 (3) , 2012
3. Ratika Pradhan, Shikhar Kumar, Ruchika Agarwal, Mohan P. Pradhan & M. K. Ghose " Contour Line Tracing Algorithm for Digital Topographic Maps" , International Journal of Image Processing (IJIP), Volume (4): Issue (2)