A MapX-based Segmentation Algorithm of Region Feature by Polyline

http://dx.doi.org/10.3991/ijoe.v9iS4.2658

Da-Sheng Wu[1,2]
[1]Zhejiang University, Hangzhou, China, [2]Zhejiang A & F University, Lin'an, Zhejiang, China

***Abstract*—MapX didn't provide a ready function which can be easy to split region feature in the client. The objective of this study was to design a segmentation algorithm for cutting region feature with polyline. In order to convenience the description, during the segmentation, only 2 intersection points were taken into consideration. According to the order of P1(the first intersection point) and P2(the second intersection point) in the R(region feature) and L(polyline feature), 4 kinds of situations had been taken into account, those were respectively, P2 was always after P1 in the R and L, P2 was after P1 in the R but P2 was before P1 in the L, P2 was always before P1 in the R and L, and P2 was before P1 in the R but P2 was after P1 in the L. Segmentation results showed that the algorithm was stable and reliable.**

***Index Terms*—MapX, Forest resource, Region feature, Segmentation algorithm.**

I. INTRODUCTION

GIS(Geographic information system) had been more widely used to forest resources management along with the continuous development of "3S" (RS, GIS, GPS) and other relational hardware and software technologies[1-5]. To some extent, as MapInfo, accepted by many users because of its simplicity and ease of use with more comprehensive functionality[6-7]. Of course, no matter what type of GIS software, considering the functional versatility, for general users, its operation was still too complex, which provided a market space for developing specialized geographic information system by programming with object-oriented programming language and GIS middleware.

Sub-compartment was the basic unit for forest resource management, and its area should be adjusted according to different management objective. Usually, in forest resources GIS, a sub-compartment should be represented as a region feature and its size should be adjusted by splitting or merging. The region feature, stored and displayed as a form of vector data , was composed of a series of ordered points, and each point was represented by one pair of coordinates (x, y). Specifically, the first point and the last point was the same point in region feature.

MapX was a kind of middleware technology that can provide most of the function of MapInfo, and it was frequently used in programming for specialized GIS[8-11]. But it did not provide a ready feature segmentation algorithm which could be easy to split dynamically sub-compartment in the client. Considering that the boundary of the region feature was usually composed by polyline with a lot of points, although previous study had described segmentation algorithm about region feature divided by straight-line[12], it should be improved.

To solve this problem, this paper designed a MapX-based segmentation algorithm of region feature by polyline, and it was programmed by VB6.0.

II. DESCRIPTION OF THE PROBLEM

Supposing that R should be segmented into R1 and R2, Pts1 and Pts2 were the ordered point sets about R1 and R2 respectively, L was a polyline, L and R intersected in 2 points of P1 and P2 whose coordinates were P1.X and P1.Y in P1, and P2.X and P2.Y in P2, meanwhile, P1.X <= P2.X.

According to the order of P1 and P2 in the R and L, generally there were 4 segmentation as following

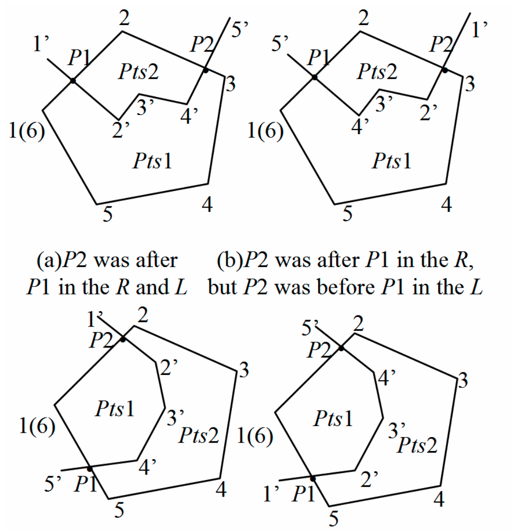
(1) As shown in Fig.1(a), P2 was always after P1 in the R and L. Where, Pts1={1,P1,2',3',4',P2,3,4,5} and Pts2={P1,2,P2,4',3',2' }.

(2) As shown in Fig.1(b), P2 was after P1 in the R, but P2 was before P1 in the L. Where, Pts1={1,P1,4',3',2',P2,3,4,5} and Pts2={P1,2,P2, 2',3',4'}

(3) As shown in Fig.1(c), P2 was always before P1 in the R and L. Where, Pts1={1,P2, 2',3',4',P1} and Pts2={P2,2,3,4,5, P1,4',3',2'}

(4) As shown in Fig.1(d), P2 was before P1 in the R, but P2 was after P1 in the L. Where, Pts1={1,P2, 4',3',2',P1} and Pts2={P2,2,3,4,5, P1,2',3',4'}

In particular, when the P1 or P2 coincides with a vertex of R, just only one of them should be added to the Pts1 and Pts2.



(a)P2 was after P1 in the R and L (b)P2 was after P1 in the R, but P2 was before P1 in the L

(c) P2 was before P1 in the R and L (d) P2 was before P1 in the R, but P2 was after P1 in the L

Figure 1. The 4 kinds of segmentation

III. ALGORITHM DESIGN

Now supposing that there were n vertices, and n-1 edges on the region feature of R, and Ej（1≤j≤n-1） was an edge which had 2 endpoint of Pj and Pj+1, Pj indicated the jth vertex; ptsChange was a boolean variable whose value could decide the points of R and L to be added to Pts1 or Pts2.

(1)Set ptsChange to false

(2)Looped for each edge of Ej, and the value of j was from 1 to n-1. Where, a virtual rectangular should be created which took Ej as main diagonal or sub-diagonal, minX and maxY as the left-top corner coordinates， maxX and minY as the right-bottom corner coordinates correspondingly. The P1 was located in Ej if and only if minX ≤ P1.X ≤maxX and minY ≤ P1.Y ≤ maxY, that meant the location of P1 in R was found. Similarly, it should be found that the position of the point of P2 located in R also.

(3)Similarly, they could be found that the position of the points of P1 and P2 located in L.

(4)Obtained the point sets of Pts1 and Pts2, and their process detailed in the program flow diagram (shown in Fig.2).

(5)Converted the point sets of Pts1 and Pts2 into independent region features which should be added into current layer immediately.

(6)Delete the polyline feature of L and the region feature of R.

IV. ALGORITHM IMPLEMENTATION

The following algorithm was based on the programming environment of Visual Basic 6.0 and MapX5.0.

```
' Variable definitions were omitted
' Find the points which located between P1 and P2 in L
 For j = 1 To L.Parts(1).Count - 1
  ' Find the internal minimum rectangular between the points of L.P(j) and L.P(j+1)
  maxX = L.Parts(1)(j).X: minX = L.Parts(1)(j + 1).X
  If L.Parts(1)(j).X <= L.Parts(1)(j + 1).X Then
    minX = L.Parts(1)(j).X: maxX = L.Parts(1)(j + 1).X
  End If
  maxY = L.Parts(1)(j).Y: minY = L.Parts(1)(j + 1).Y
  If L.Parts(1)(j).Y <= L.Parts(1)(j + 1).Y Then
    minY = L.Parts(1)(j).Y: maxY = L.Parts(1)(j + 1).Y
  End If
  If P1.X >= minX And P1.Y >= minY And P1.X <= maxX And P1.Y <= maxY Then
    StartPNo = j
  End If
  If P2.X >= minX And P2.Y >= minY And P2.X <= maxX And P2.Y <= maxY Then
    EndPNo = j
  End If
 Next
 myStep = 1
 If StartPNo > EndPNo Then
  myStep = -1
  EndPNo = L.Parts(1).Count - EndPNo: StartPNo = L.Parts(1).Count - StartPNo + 1
 Else
  StartPNo = StartPNo + 1
 End If
 P2afterP1 = 0
 For j = 1 To R.Parts(1).Count - 1
  ' Find the internal minimum rectangular between the points of R.P(j) and R.P(j+1), and codes are skipped
  ' Process the common points
  If Not ((P1.X = R.Parts(1)(j).X And P1.Y = R.Parts(1)(j).Y) Or (P2.X = R.Parts(1)(j).X And P2.Y = R.Parts(1)(j).Y)) Then ' When the intersection points do not coincide with the points of R.P(j)
   If Not ptsChange Then
    Pts1.Add R.Parts(1)(j)
   Else
    Pts2.Add R.Parts(1)(j)
   End If
  End If
  ' Process the intersection point of P1
  If P1.X >= minX And P1.Y >= minY And P1.X <= maxX And P1.Y <= maxY Then
   Pts1.Add P1: Pts2.Add P1: P2afterP1=P2afterP1+ 1
   If myStep=1 Then 'when the P1 is before P2 in L
   If P2afterP1=1 Then 'when the P1 is before P2 in R
     For k = StartPNo To EndPNo Step myStep
      Pts1.Add L.Parts(1)(k)     ' add the points to Pts1
     Next
    Else ' when the P1 was after P2 in the L
     For k = StartPNo To EndPNo Step myStep
      Pts2.Add L.Parts(1)(k)     ' add the points to Pts2
     Next
    End If
   Else ' when the P1 was after P2 in the R
    If P2afterP1 = 1 Then
     For k = EndPNo To StartPNo Step myStep
      Pts1.Add L.Parts(1)(k)     ' add the points to Pts1
     Next
    Else
     For k = EndPNo To StartPNo Step myStep
      Pts2.Add L.Parts(1)(k)     ' add the points to Pts2
     Next
    End If
   End If
   If P1.X = R.Parts(1)(j).X And P1.Y = R.Parts(1)(j).Y Then  'when P1=R.P(j)
    j = j + 1
   End If
   ptsChange = Not ptsChange
  End If
 ' Considering its processing similarity to the intersection point of P1, the process of P2, its codes is also skipped.
```

```
Next
' Process the last point of R
If lastPoint = P1 Or lastPoint = P2 Then
 Pts1.Add R.Parts(1)(j): Pts2.Add R.Parts(1)(j)
Else
 If Not ptsChange Then
  Pts1.Add R.Parts(1)(j)
 Else
  Pts2.Add R.Parts(1)(j)
 End If
End If
Set R1 = FtrFac.CreateRegion(Pts1, R.Style.Clone): Set R2 = FtrFac.CreateRegion(Pts2, R.Style.Clone)
Set ftrnew1 = lyr.AddFeature(R1): Set ftrnew2 = lyr.AddFeature(R2)  "add the R1 and R2 into the layer of lyr
lyr.DeleteFeature R: lyr.DeleteFeature  ' delete the old features of R and L
```

V. THE SIMULATION RESULTS

As shown in Fig.3, the simulation results indicated that the algorithm had achieved to the segmentation to region feature based on polyline.

VI. CONCLUSION

MapX did not provide a ready segmentation algorithm which could be easy to split region feature in the client dynamically. To solve this problem, this paper proposed a MapX-based segmentation algorithm of region feature by polyline. The algorithm had taken a variety of complex segmentation into account based on 2 intersection points. Segmentation results showed that the algorithm was stable and reliable.

ACKNOWLEDGMENT

This work was financially supported by Zhejiang province major science and technology project (2011C12047) and Zhejiang province education department project(Y201225717).

REFERENCES

[1] Chu, L.X., Zhang, L., "Design of forest resources data system based on GIS database", *Journal of Central South University of Forestry & Technology*, vol.32, no.6, pp.48-54, 2012.

[2] Wu, L.C., Li, C.G., Chen, H.A., "Design and implementation of the universal forestry resource survey GIS", *Science of Surveying and Mapping*, vol.36, no.4, pp.255-257, 2011.

[3] Li, X.D., Zhang, Z., "The development of GIS-based archive management system of forest farm resources", *Journal of Changchun University*, vol.20, no.12, pp. 69-73, 2010.

[4] Singh, I.J., Jugran, D.K., Thanruma, S., Reddy, S.R., "Forest resource assessment in mohand forest range, Uttar Pradesh using remote sensing and GIS", *Journal of the Indian Society of Remote Sensing*, vol.33, no.4, pp.565-574, 2005. http://dx.doi.org/10.1007/BF02990741

[5] McCall, M.K., Minang, P.A., "Assessing participatory GIS for community-based natural resource management: Claiming community forests in Cameroon", *Geographical Journal*, vol.171, no.4, pp.340-356, 2005. http://dx.doi.org/10.1111/j.1475-4959.2005.00173.x

[6] Huang, S.C., Xu K.J., "Establishment and Application of GIS of Forest Resources at County Level Based on Mapinfo", *Journal of Guangxi Academy of Sciences*, vol.22 no.4, pp.333-336, 2006.

[7] Xiao, W., "Forest Distribution Mapping of Hubei Province based on MapInfo Software", *Forest Inventory and Planning*, vol.31, no.2, pp.17-20, 2006.

[8] Zhang, Q., Yin, X., "Best path analysis in military highway transportation based on MapX", *Advances in Intelligent and Soft Computing*, vol.129, pp. 163-168, 2011. http://dx.doi.org/10.1007/978-3-642-25986-9_25

[9] Liu, X.D., Chen, Y. W., Long, Y.J., "A MapX-based preprocessing approach for multi-satellite cooperative observation towards area target", *System Engineering Theory and Practice*, vol.30, no.12, pp.2269-2275, 2010.

[10] Wu, J., Niu, D., Liu, S., "Design and implementation of rainfall retrieval system of weather radar based on MapX", *Journal of Geomatics*, vol.35, no.3, pp.19-20, 2010.

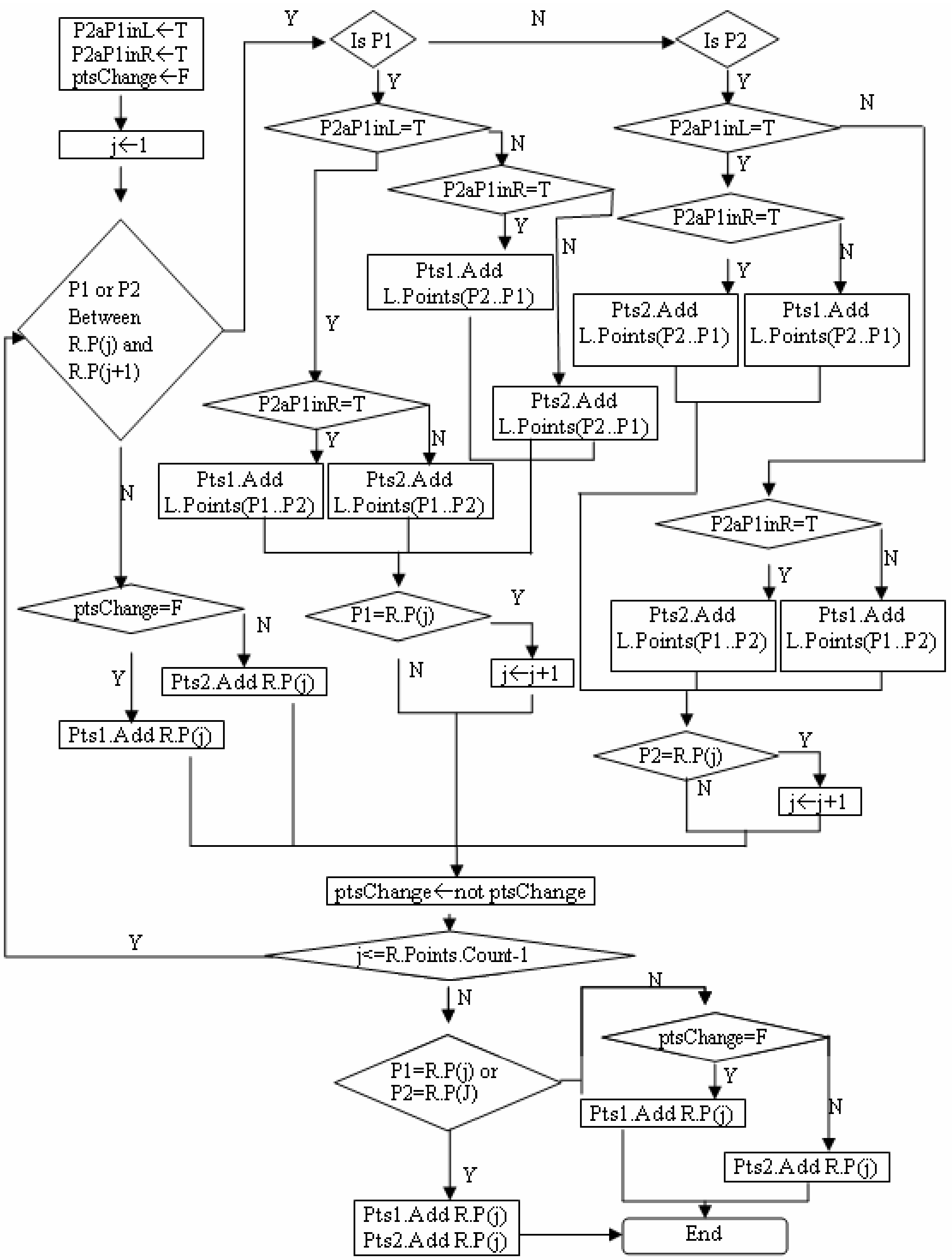
[11] Yan, P., Wu, W., Wang, X., "Pilot design of biomass energy resources inquiry system based on MapX: A case of Liyang city in China", *International Journal of Global Energy Issues*, vol.29, no.3, pp.294-302, 2008. http://dx.doi.org/10.1504/IJGEI.2008.018010

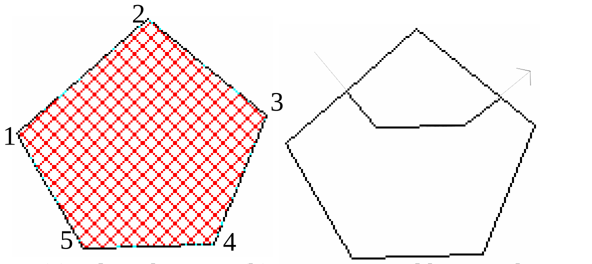
[12] Wu, D.S., Feng, X., Wen, Q.Q., "The segmentation algorithm of region feature based on MAPX", *Advances in Intelligent and Soft Computing* 105(2011) 657-661. http://dx.doi.org/10.1007/978-3-642-23756-0_105

AUTHORS

Da-Sheng Wu is with Institution of Remote Sensing and Information System Application, Zhejiang University, Hangzhou 310029, China, Tel: 1-396-802-3105, (E-mail: wu62390710@263.net).

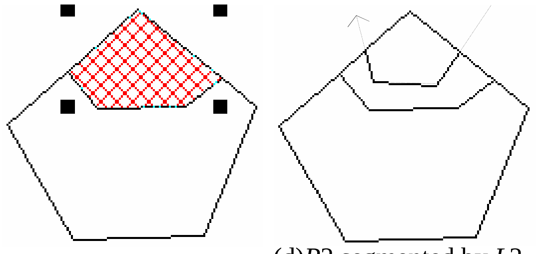
This article is an extended and modified version of a paper presented at the International Conference on Mechanical Engineering, Automation and Material Science (MEAMS2012), held 22-23 December 2012, Wuhan, China. Received 08 April 2013. Published as resubmitted by the authors 01 May 2013.


P2aP1inL←T
P2aP1inR←T
ptsChange←F
j←1
P1 or P2 Between R.P(j) and R.P(j+1)
Is P1
Is P2
P2aP1inL=T
P2aP1inR=T
Pts1.Add L.Points(P2..P1)
Pts2.Add L.Points(P2..P1)
Pts1.Add L.Points(P1..P2)
Pts2.Add L.Points(P1..P2)
ptsChange=F
Pts2.Add R.P(j)
Pts1.Add R.P(j)
P1=R.P(j)
j←j+1
P2=R.P(j)
ptsChange←not ptsChange
j<=R.Points.Count-1
P1=R.P(j) or P2=R.P(J)
Pts1.Add R.P(j)
Pts2.Add R.P(j)
End
Y
N



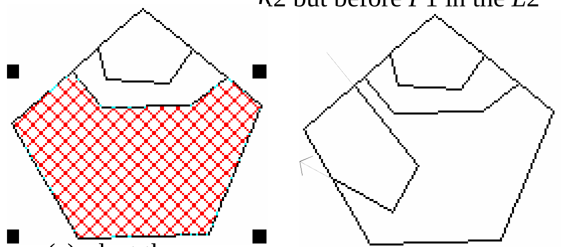
(a) select the feature of *R*1

(b)*R*1 segmented by *L*1 when *P*2 was after *P*1 in the *R*1 and *L*1



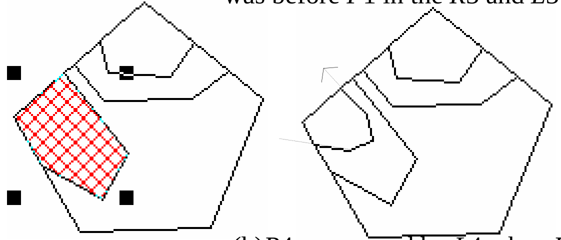
(c)select the feature of *R*2

(d)*R*2 segmented by *L*2 when *P*2 was after *P*1 in the *R*2 but before *P*1 in the *L*2



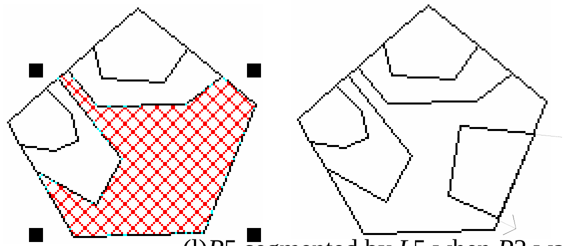
(e)select the feature of *R*3

(f)*R*3 segmented by *L*3 when *P*2 was before *P*1 in the *R*3 and *L*3



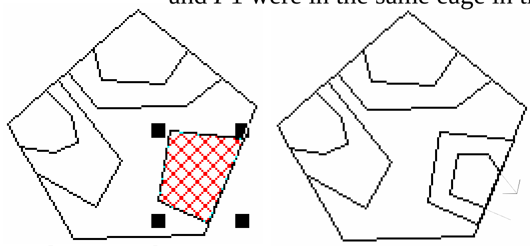
(g)select the Feature of *R*4

(h)*R*4 segmented by *L*4 when *P*2 was before *P*1 in the *R*4 but was after *P*1 in the *L*4



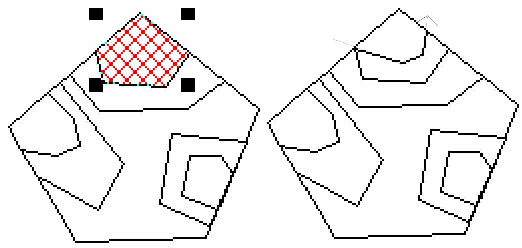
(k)select the feature of *R*5

(l)*R*5 segmented by *L*5 when *P*2 was before *P*1 in the *R*5 and *L*5, meanwhile *P*2 and *P*1 were in the same eage in the *R*5



(m)select the feature of *R*6

(n)*R*6 segmented by *L*6 when *P*2 was before *P*1 in the *R*6 but after *P*1 in the *L*6, meanwhile *P*2 and *P*1 were in the same eage in the *R*6



(m)select the feature of *R*7

(n)*R*7 segmented by *L*7 when *P*2 was after *P*1 in the *R*7 and *L*7 , meanwhile *P*1 coincided with a vertex of the *R*7

Figure 3. The Result of Segmentation