PhD Showcase: A Model for Progressive Transmission of Spatial Data Based on Shape Complexity

PhD Student: Fangli Ying
Department of Computer Science
NUI Maynooth
Co. Kildare. Ireland
fying@cs.nuim.ie

PhD Supervisor: Peter Mooney[*]
Environmental Research Center
Environmental Protection Agency
Clonskeagh,Dublin 14. Ireland
p.mooney@epa.ie

PhD Supervisor: Padraig Corcoran
Department of Computer Science
NUI Maynooth
Co. Kildare. Ireland
padraigc@cs.nuim.ie

PhD Supervisor: Adam C. Winstanley
Department of Computer Science
NUI Maynooth
Co. Kildare. Ireland
adamw@cs.nuim.ie

ABSTRACT

Due to the limited bandwidth available to mobile devices transmitting large amount of geographic data over the Internet to these devices is challenging. Such data is often high-resolution vector data and is far too detailed with respect to most location-based services (LBS) user requirements. A less detailed version may be sent prior to the complete dataset using a progressive transmission strategy. Progressive transmission is generally performed by transmitting a series of independent pre-computed representations of the original dataset at increasing levels of detail where the transitions between these levels are not necessarily smooth. A model is proposed in this paper for selective progressive transmission which will provide smoother transmission over increasing levels of detail. We define criteria for the comparison of similarity between the progressive states of the vector-data based on shape complexity of the polygon features. This allows development of a real-time strategy for the progressive transmission of vector data over the Internet to mobile devices.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Spatial databases and GIS; H.3.5 [**Online Information Services**]: Data sharing

General Terms

Human Factors

Keywords

Progressive Transmission, Vector data, Shape Complexity OpenStreetMap

1. INTRODUCTION

Making publicly available geographic datasets available for users to view, download and analyse over the Internet is now an important topic in web-GIS and Location-based Services (LBS). However the increasing amounts of data coupled with sometimes slow communication links to Internet-enabled mobile devices means transmitting such large amounts data over the Internet is often difficult. For example when a user with a mobile device is attempting to download and visualize a large amount of spatial data there are a number of user interaction issues which include the problem of displaying a large amount of spatial data on a small screen and the need to prevent the user from having to wait a long time to receive the full map representation. Progressive transmission is a promising technique to address these practical problems in these situations. This PhD in Computer Science commenced in October 2009. This paper describes the current state of progress of the research work. The aim of the reesarch work is to improve the user experience for users of mobile devices accessing Location-based Services (LBS) when downloading and visualising spatial data on these small-screen devices. Some practical examples of location-based applications for our proposed model include: tourist maps of wildlife areas; environmental monitoring maps; and pedestrian navigation. Our paper is organised as follows. In section 2 we give an overview of related work on progressive transmission both in GIS and other computing disciplines. To give an overview of the types of OpenStreetMap data used in the development of this model we describe the processing of OpenStreetMap XML data in section 3. The detailed description of our model for progres-

---

[*]Also at: Dept of Comp Sci, NUI Maynooth, Ireland.

The SIGSPATIAL Special, Volume 2, Number 3, November 2010.

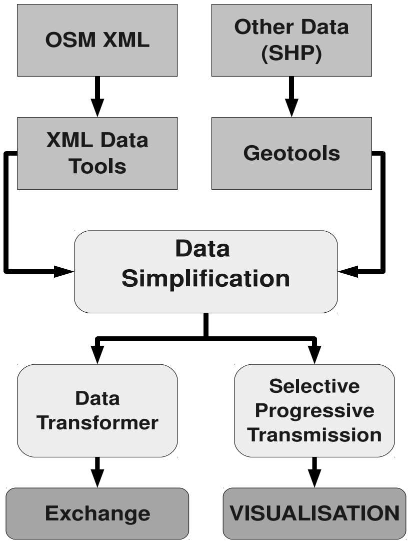


Figure 1: A flowchart of components in this selective progressive transmission

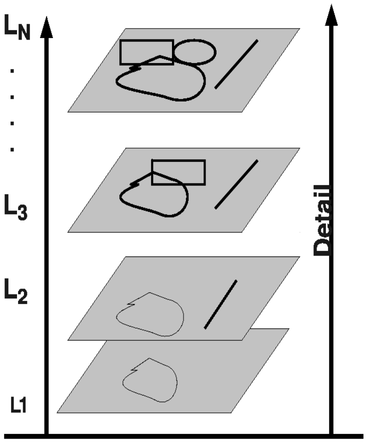


Figure 2: A hypothetical example of selective progressive transmission where level L_N is the original highest level of detail and L_1 is the most simplified version of the original L_N

sive transmission based on shape complexity begins in section 4 with a detailed description of polygon simplification with emphasis on shape preservation. Selective progressive transmission based on shape complexity rules is described in section 5. The paper closes with section 6 where we provide some initial results of implementation of our model, discussion of the development of the model, and our plan for immediate and long-term future work in this area.

2. OVERVIEW OF RELATED RESEARCH

Progressive transmission usually provides a number of precomputed levels of representation of a spatial dataset which can be delivered quickly to meet the time requirements of users. However these pre-computed levels are computed offline and may not be updated regularly. Given the size of some spatial datasets it may take a large amount of time to update these levels. Several authors have highlighted problems with this approach with Jones and Ware [6] stating that these multiple representations may differ markedly in their degree of generalization while the size limitations of mobile devices make it all the more desirable that the level of generalisation to be adapted flexibly to meet the needs of individual users. The use of progressive transmission of spatial data would be much more flexible if it could be performed in real-time and be adapted to the current user's needs and spatial location. Unfortunately there has been very little work carried out on the progressive transmission of spatial data. It is necessary to look to the domain of computer graphics where similar problems also arise in the Levels of Detail (LOD) approximation for the progressive transmission of high detailed geometric models. Lounsbery *et al* [12] proposed the concept of multi-resolution analysis to surfaces of arbitrary topological type. Since real-time switching between LOD for the meshes representing these geometric models may lead to perceptible "popping" effect the goal is to construct a progressive transmission model which has smooth visual transitions between meshes at different resolutions. Eck *et al* [4] describes a series of efficient strategies for progressive transmission of meshes and using selective refinement to optimize the LOD representations. In related work Hoppe [5] proposed an efficient algorithm for selective refinement for incrementally adapting the mesh refinement in order to reduce "popping" effects. This approach is also invertible whereby the progressively transmitted levels of meshes can deconstructed for any level of resolution. In GIS progressive transmission in remains a challenging problem because of the intrinsic complexity of map generalization [1]. One of the most challenging aspects of this part of progressive transmission is topologically consistent generalization methods. The standard method for progressive transmission is to generalized data into a series level of details for incrementally delivery (Lehto). Most readers will be familiar with this concept from the use of map-tile based web mapping such as Google Maps or Bing Maps. While this standard method for progressive tranmission offers multiple-resolution views of the spatial data the user experience can be effective by the similar problem of "popping" mentioned above. This "popping" in progressive transmission of spatial data occurs because of the difference of representations among the pre-computed levels for the given spatial datasets. The naive solution would be to precompute enough levels of representation such that there is a smooth visual transition between levels. However this is completely impractical because for the more levels of resolution required the greater the storage requirements on the server side would be. This would also result in the same data being sent multiple times. The concept of "popping" is best illustrated from the example in Figure 2 where the distance between successive levels is too great (like that between L_2 and L_3) and information appears to "pop" into the current representation. Map generalization techniques are generally performed in progressive transmission for generating a coarser representation of the data before the remainder of data is incrementally transmitted. There are several solutions for automated line generalizations which have already been defined with respect to several constraints. For example some methods of line simplification were developed to ensure topological consistency [14]. Other methods are intended to preserve shape characteristics [16].

3. PROCESSING OSM-XML DATA

Before we describe how to process the OSM-XML we give a brief overview of the use-case scenario for progressive transmission. A user selects an area from a web-based map displaying OpenStreetMap data. Using the OSM API the OSM XML data corresponding to the area selected is downloaded in real-time. Using the Stax XML Java toolkit the OSM XML is processed. The geographic objects (points, polygons, polylines) are then bound to Java objects. The spatial data then undergoes simplification from the highest level of detail (L_n) to the most simplified version (L_1). The order in which the objects are simplified is maintained using a set of arrays which hold the Java objects containing the spatial data. When simplification is completed the data is then progressively transmitted to the mobile device by progressively transmiting the data out to the mobile device from version L_1 to L_n. As described in the flowchart in figure 1 the data transformer component can also deliver *packages* of the spatial data at any of the levels of simplification from L_1 to L_n in OSM XML format or other vector data formats. OpenStreetMap XML (OSM-XML) is one of several formats in which the raw OSM geographic data is made publicly available for download. Most LBS enabled mobile devices are still unable to handle vector data delivered in XML-based formats such as OSM XML. The problem is compounded by the fact that very often the OSM XML can represent a very large geographical area and/or contains a very large number of geographic objects. OSM XML contains point, line and polygon features. Every spatial attribute (or tag) corresponding to each feature is included in the OSM-XML. Very often the size of OSM-XML files corresponding to very small geographical areas in locations where OpenStreetMap coverage is very good can be several megabytes in size. As illustrated in Figure 1 in section 1 the OSM-XML is downloaded in real-time and processed using an Open Source XML data processing framework called Stax which is suitable for processing XML data using streaming. Stax allows the stream-based processing of the OSM XML files to the mobile device in real-time. The OSM API `http://wiki.openstreetmap.org/wiki/API_v0.6` is used for on-the-fly XML data capture. We can introduce constaints upon the size and extent of the geographical data which the user can select for download and visualisation.

Considering that most LBS-enabled devices have limited storage and computing ability the optimized streaming-based XML API such of Stax is alternative approach for real time pre-processing OSM-XML. Unlike traditional tree-based tools or pure streaming-based tools, such as JDOM or SAX, for XML processing Stax has many advantages. These advantages include the availability of cursor level access to the XML data and effecient memory management techniques. The advantage of Stax is that we can easily move the cursor pointer forward, skip to any specified geographic feature in the XML, and finally extract the OSM data efficiently without prohibitive memory consumption. The software implementation of our model for progressive transmission is written in Java. This Java-based approach means that our application will run on most Java-enabled mobile devices. Moreover, when presented with raw geographic data in some vector data formats, such as ESRI shapefile, the Java Geotools library, is a very useful tool for transforming and processing these data formats. Currently we are using the OpenStreetMap (OSM) database as a case study dataset. However with the use of the Java GeoTools library the model for progressive transmission described in this paper can access any well known vector data format. Figure 1 shows a flowchart of our proposed model for selective progressive transmission.

4. SHAPE SIMPLIFICATION WHILE PRESERVING POLYGON CONTOUR CHARACTERISTICS

The screen displays on LBS-enabled devices are usually small and of relatively low resolution. This means that visualisation is not comparable to a map presentation in a GIS or within a desktop application [15]. It is most important to preserve contour shape without an over-represented of detail. Most polygons can be generalised and simplified as they have nodes which can be removed without adversely affecting the overall shape of the polygon and how the shape is interpreted by the human visual system [9, 10]. Informally a polygon can undergo simplification if the removal of a subset of the polygon nodes can be performed without affecting the overall shape of the polygon to such an extent that it is unrecognisable from its original form. Only insignificant vertices can be considered for removal during simplification. Latecki et al. [11] proposed the following metric K which determines the significance of each vertex to the overall shape of the polygon in question. Suppose for some vertex s in the polygon p with incident edges on s called s_1 and s_2 then the K metric for significance is given by:

$$K\left(s_1, s_2\right) = \frac{\beta\left(s_1, s_2\right) l\left(s_1\right) l\left(s_2\right)}{l\left(s_1\right) + l\left(s_2\right)}. \qquad (1)$$

Where l is the length function normalized with respect to the total contour length of the polygon, and $\beta\left(s_1, s_2\right)$ is the turning angle at the vertex in question. Informally this metric will determine vertices with a greater turning angle and adjacent edges of a greater length as being most significant. The effectiveness of this metric at determining vertex significance is demonstrated by Figure 3 where polygon vertices are highlighted in two polygons A and B. The removal of the vertex circled in polygon B in Figure 3 would dramatically alter the overall shape of the contour in question. This node receives a more significant corresponding K value. On the other hand the vertices circled in polygon A could be removed without any significant changes to the overall shape of the contour of polygon A and are assigned an insignificant K value very close to zero.

4.1 Simplification of Polygon Shapes

For a set of polygons P the following algorithm is applied to each polygon p in the set. The establishment of the λ parameter (by trail and error) allows the simplification of all polygons in P to a similar resolution. Over-represented polygons will undergo more steps of simplification than other most suitably represented polygons. To establish the overall significance KS of removing nodes from a given polygon p the following steps are performed:

1. For each polygon node with adjacent edges i and j, determine its corresponding significance by evaluating $K(i,j)$.
2. Calculate $Kmean$ which represents the mean of all K over all polygon nodes; that is $Kmean = \frac{\sum K(i,j)}{N}$ where N is the number of nodes in the polygon p.
3. If $Kmean > \lambda$ where λ is a predefined threshold then this polygon p is not simplified further as the vertices are all significant. The polygon p is a candidate for direct delivery. Otherwise go to the next step.

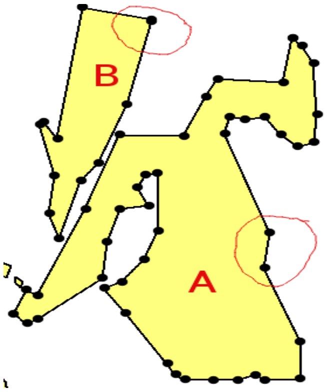


Figure 3: Example of vertex significance for two polygons

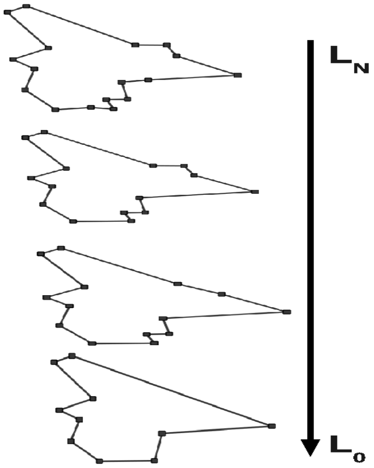


Figure 4: Node Reduction Example

4. Extract the node $n_{i,j}$ in polygon p with minimum $K(i,j)$. This node is removed from the polygon p and placed on top of the node stack.

5. Update the polygon p data structure - node $n(i,j)$ has been removed. The incident nodes x and y to n through edges i and j become connected directly by a new edge ij. Recalculate K for both x and y nodes. Recalculate the $Kmean$ value. Go to step 3.

The illustration in Figure 4 shows a simple example of this algorithm applied to a single polygon. The polygon is presented at full resolution at level L_N. At each level from L_N down to L_0 the lowest level of simplification the nodes with the minimum $K(i,j)$ are removed. At level L_0 the $Kmean > \lambda$ and the simplification of this polygon is stopped. Each polygon p in the set of polygons P has a corresponding node stack. This is an array data structure which holds the node objects in sorted order. Each node object n has the following attributes: K value, integers to indicate the nodes that it is connected to in p, an index r which indicates its position in the polygon p, and a Point object which holds its geographical coordinates. In the Java implementation of this model we have implemented polygons and nodes as `Comparable` objects. These objects can be stored and managed by Java's Collections Framework which offers several efficient libraries for sorting arrays and lists of `Comparable` objects.

4.2 Progressive Transmission of Contour Preserved Shapes: Implementation

As described in Section 4.1 the selected polygons are simplified according to rules which preserve the shapes of the polygons. To complete the progressive transmission of the selected polygons to the user's mobile device the process of Section 4.1 (and illustrated in Figure 4) must be reversed. The most simplified representation of the selected polygons are stored at level L_0. These polygons are delivered to the mobile device and visualised on the screen using a mapping framework such as OpenLayers. The process of progressive transmission from L_0 to full resolution at L_N works by progressively selecting and transmitting nodes from the polygon node stacks. At each iteration T most significant nodes are popped off the node stacks and transmitted. If $T = 1$ then one node is popped off. This node is the node with largest $K(i,j)$ amongst the node stacks of all polygons in P. For $T > 1$ the T largest $K(i,j)$ values are popped off the node stacks. When these T nodes are transmitted to the mobile device Javascript/AJAX functionality in the visualisation module updates the map display by updating the polygons to include the newly arrived T nodes. The progressive transmission of the polygons is completed when all of the node stacks are empty or if the user decides to request visualisation of a different geographical area.

One of the problems with this approach is that there is no consideration given to the overall shape complexity of the polygons selected. The T nodes which are progressively transmitted on each level L_i are taken from the node stacks. In the next section we described an enhanced model of selective progressive transmission which takes the shape complexity of the polygons into consideration before transmitting any nodes. Broadly speaking the polygon(s) at L_i which are most dissimiliar to their corresponding full resolution representation at L_N are selected and nodes belonging to these polygons are selectively transmitted.

5. SELECTIVE PROGRESSIVE TRANSMISSION BASED ON SHAPE COMPLEXITY

In this section we describe an enhanced version of the selective progressive transmission strategy proposed in Section 4. This version of the selective progressive transmission strategy is based on using rules generated from the shape complexity of the polygons selected by the user. By using shape complexity to calculate the dissimilarity of the polygon(s) at L_i to their corresponding full resolution representation at L_N the polygon(s) which should have nodes added are selected. This builds upon work by Joshi *et. al* [7] who describe a dissimilarity function that can be used in state-of-the-art spatial clustering algorithms. This results in clusters of polygons that are more compact in terms of spatial contiguity. The concept of spatial contiguity is very important in spatial data clustering and visualisation [13]. The dissimilarity function for the shape complexity of a polygons proposed by Joshi *et. al* [7] measures the distance between n scalar spatial attributes of the polygons. The distance can be measured using standard the Euclidean distance metric. If $n = 2$ then this is just 2 dimensional cartesian space. There have been many characteristics proposed to describe the structural shape complexity and characteristics of a polygon object. Brinkoff *et .al* [2] provide a description of the most popular shape complexity measures for spatial data polygons. In other work [19, 18] the authors use a small subset of the measures described by Brinkoff *et .al.* These are described as follows:

- Normalised Area Ratio (AR) - the area difference between the area of the polygon q ($A(q)$) object and its convex hull ($A(C(q))$) expressed as $\frac{A(C(q))-A(q)}{A(C(q))}$. An AR value of 1.0 indicates that the convex hull perfectly fits the polygon. As the value approaches 0.0 this indicates an increasingly "spiky" polygon where the convex hull is much larger than the polygon itself.

- Circularity C - an expression of the compactness of the polygon object q - $\frac{4\pi * A(q)}{P(q)^2}$ where $P(q)$ is the perimeter of the polygon object. A circularity value of 1.0 indicates a perfect circle. As the value approaches 0.0, it indicates an increasingly elongated polygon.

5.1 Progressive Transmission based on Shape Complexity: Algorithm

This method begins by calculating the Normalised Area Ratio (AR) and the Cicularity C of each of the polygons P in the geographical area chosen by the user. The polygons are stored at full resolution at L_N. The values of AR and C are stored as scalar attributes of the polygon objects. The establishment of the λ parameter is set as before. The process described in Section 4.1 is used to simplify the polygon shapes. When the simplification is completed (that is all polygons in P have reached step 3 in the process from Section 4.1) the AR and C are computed again for all of the simplified polygon objects at L_0. Now the selective progressive tranmission can begin to deliver the polygon objects back to the user device for visualisation. In the next section we describe how this algorithm is implemented. A dissimilarity measure $D(L_0^p, L_N^p)$ between a polygon p at the final level of simplifcation L_0^p and the same polygon p at full resolution representation L_N^p is given by:

$$D(L_i^p, L_N^p) = \sqrt{(AR_{L_i}^p - AR_{L_N}^p)^2 + (C_{L_i}^p - C_{L_N}^p)^2} \qquad (2)$$

The measure $D(L_0^p, L_N^p)$ allows the selective progressive transmission to add more node details to polygons which are very dissimiliar (within the parameters of $Kmean > \lambda$). The measure $D(L_0^p, L_N^p)$ is easily computed. In Ying *et. al* [19, 18] the authors show that the combination of the scalar attributes of AR and C can cluster spatial polygons into two distinct clusters: one with polygons with high $Kmean$ (complex shapes) value the other with low $Kmean$ value (simple shapes).

5.2 Progressive Transmission based on Shape Complexity: Implementation

The set of simplified polygon objects at L_0 must be progressively transmitted to the user. The next step in the progressive transmission of the vector data to the user is to select which nodes are transmitted next. The process is as follows.

- At L_0 the AR_{L_0} and C_{L_0} are computed for all of the simplified polygon objects at L_0. For each of the polygons p we calculate the dissimilarity measure $D(L_0^p, L_N^p)$ between the complexity of polygon p at L_0^p and L_N^p.

- The T most significant nodes are popped off the node stack for the polygon p with the largest dissimilarity measure $D(L_0^p, L_N^p)$. If $T = 1$ then one node is popped off. This node is the node with largest $K(i, j)$ amongst the node stacks of all polygons in P. For $T > 1$ the T largest $K(i, j)$ values are popped off the node stacks. When these T nodes are transmitted to the mobile device Javascript/AJAX functionality in the visualisation module updates the map display by updating the polygons to include the newly arrived T nodes.

- The steps 1 and 2 are repeated for subsequent levels L_k to L_N. The parameter T can be adjusted to control the number of levels from L_k to L_N

The dissimilarity measure $D(L_i^p, L_N^p)$ could be extended to use additional scalar shape complexity attributes such as those proposed by Brinkhoff *et al*[2]: notches (number of large turning angles in the polygon), polygon convexity measurement, or perimeter ratio.

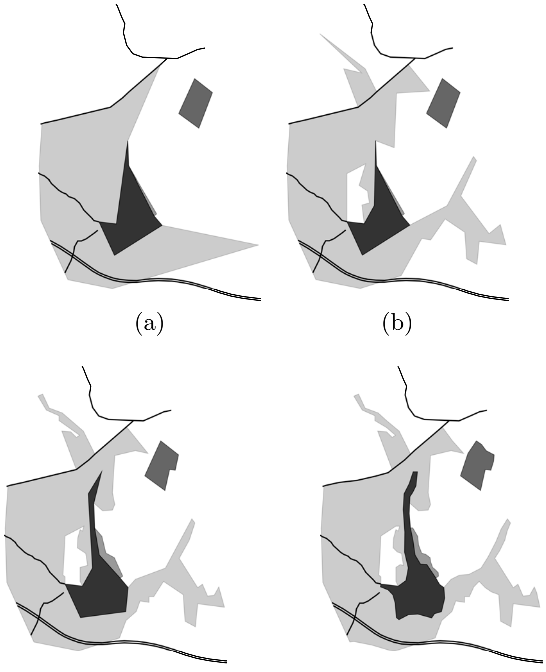
6. DISCUSSION AND CONCLUSIONS

The initial work of an early stage PhD in Computer Science is described. A model for progressive transmission of spatial data based on shape complexity has been proposed in this paper. The target area of implementation is in the delivery of spatial data to mobile devices accessing Location Based Services (LBS). As described in the literature review of Section 2 progressive transmission of GIS data is a difficult problem. Our model aims to tackle some of these problems most notably the issue of providing a smooth transmission between *data levels* or representations. Currently our model only simplifies polygons. Polylines are transmitted without simplication. We are currently working on including progressive transmission of polylines also using the Doughas Peucker algorithm for polyline simplification [3]. An issue for further work is comparing our progressive transmission model with compression techniques for spatial data. Several vector-data compressive techniques have been proposed recently namely variable-resolution compression [17] and algorithms for lossy vector data compression [8]. Both techniques demonstrate a feasible and efficient solutions for the compression of vector data, are able to achieve good compression ratios and maintains the main shape characteristics of the spatial objects within the compressed vector data.

Figure 5a to 5d displays some sample steps in the progressive transmission of a set of polygons. The simplified representation of the polygons in question is displayed in Figure 5a and contains 38 polygon vertices in total. Detail is progressively added to this representation in Figure 5b and Figure 5c which contain 70 and 97 polygon vertices respectively. The progressive transmission process completes when the entire data set has been received and integrated as shown in Figure 5d. This final map contains a total of 182 polygon vertices. This example shows the very initial results of the proposed progressive transmission strategy. There are a number of potential practical implementations of progressive transmission of vector data to mobile devices. For example in environmental monitoring a user can move quickly through an area and may stop to make samples or measurements. Not all high resolution data is required at all times and progressively more detailed data can be delivered and visualised depending on the user's requirements.

Acknowledgments

Fangli Ying is funded by Irish Universities Association and China Scholarship Council. The Environmental Protection Agency provides funding to Dr. Peter Mooney grant (2008-FS-DM-14-S4). Research presented in this paper was also funded by a Strategic Research Cluster grant (07/SRC/I1168) by Science Foundation Ireland under the National Development Plan. The authors gratefully acknowledge these supports.


(a) (b)

(c) (d)

Figure 5: A sample of steps in the progressive transmission of a set of polygons are shown.

7. REFERENCES

[1] M. Bertolotto and M. Egenhofer. Progressive transmission of vector map data over the world wide web. *GeoInformatica*, 5(4):345–373, 2001.

[2] T. Brinkhoff, H. Kriegel, and R. Schneider. Comparison of approximations of complex objects used for approximation-based query processing in spatial database systems. In *The Ninth IEEE International Conference on Data Engineering*, Vienna, Austria, Apr. 1993.

[3] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, Dec. 1973.

[4] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. In R. Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 173–182. ACM SIGGRAPH, Addison Wesley, Aug. 1995. held in Los Angeles, California, 06-11 August 1995.

[5] H. Hoppe. Progressive meshes. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 99–108, New York, NY, USA, 1996. ACM.

[6] C. B. Jones and J. M. Ware. Map generalization in the web age. *International Journal of Geographical Information Science*, 19(8-9):859–870, 2005.

[7] D. Joshi, A. Samal, and L.-K. Soh. A dissimilarity function for clustering geospatial polygons. In *GIS '09: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 384–387, New York, NY, USA, 2009. ACM.

[8] A. Kolesnikov. Optimal algorithm for lossy vector data compression. In M. Kamel and A. Campilho, editors, *Image Analysis and Recognition*, volume 4633 of *Lecture Notes in Computer Science*, pages 761–771. Springer Berlin / Heidelberg, 2007.

[9] L. Latecki and R. Lakamper. Polygon evolution by vertex deletion. pages 398–409, 1999.

[10] L. J. Latecki and R. Lakaemper. Contour-based shape similarity. *Lecture Notes in Computer Science*, 1614:617–630, 1999.

[11] L. J. Latecki and R. Lakmper. Convexity rule for shape decomposition based on discrete contour evolution. *Computer Vision and Image Understanding*, 73(3):441 – 454, 1999.

[12] M. Lounsbery, T. DeRose, and J. Warren. Multiresolution analysis for surfaces of arbitrary topological type. *ACM Transactions on Graphics. Also: Technical Report, 93-10-05b, Dept. of Comp. Sci., Univ. of Washington*, 1997.

[13] R. T. Ng and J. Han. Clarans: A method for clustering objects for spatial data mining. *IEEE Trans. on Knowl. and Data Eng.*, 14(5):1003–1016, 2002.

[14] S. Z. P. M. van der Poorten and C. B. Jones. Topologically-consistent map generalisation procedures and multi-scale spatial databases, 2002.

[15] M. Sester and C. Brenner. Continuous generalization for visualization on small mobile devices. In P. Fisher, editor, *11th International Symposium on Spatial Data Handling*, Developments in Spatial Data Handling, pages 355–368. Springer Berlin Heidelberg, August 2005.

[16] Z. Wang and J.-C. Muller. Line generalization based on analysis of shape characteristics. *Cartography and Geographic Information Science*, pages pp. 3–15(13), January 1998.

[17] B. Yang, R. S. Purves, and R. Weibel. Variable-resolution compression of vector data. *Geoinformatica*, 12(3):357–376, 2008.

[18] F. Ying, P. Mooney, and P. Corcoran. Using shape complexity to guide simplification of geospatial data for use in location-based services. In G. Gartner and M. LIU, editors, *The 7th International Symposium on LBS & TeleCartography*, page (To appear), Hidleberg, Germany, September 2010. Springer - Lecture Notes in Computer Science.

[19] F. Ying, P. Mooney, P. Corcoran, and A. Winstanley. Polygon processing on openstreetmap xml data. In M. Haklay, J. Morely, and H. Rahemtulla, editors, *Proceedings of the GIS Research UK 18th Annual Conference*, pages 149–154, London, England, 2010. University College London.