GEOMETRY UPDATING FOR GEOSPATIAL DATA INTEGRATION

Ismail Wadembere, Patrick Ogao
Faculty of Computing & Information Technology, Makerere University
P.O. Box 7062, Kampala, Uganda
iwadembere@cit.mak.ac.ug, ogao@cit.mak.ac.ug

Abstract

Lots of geospatial data has been collected within the last decades due to advances in digital spatial data capturing technologies and this has brought in different versions of the same data sets. Although various spatial and mapping organizations are updating and revising geospatial databases with new data/information about our rapidly changing environment; there are challenges of how to effectively and quickly update and adjust the geometries so that there are no sliver and dangling issues as a result of opening and overlaps due to variation between different versions of same data. This is vital in geospatial data updating and management in geo-systems and databases. The paper describes an approach of geospatial geometry updating and adjustment basing on paradigm of point as being the simplest and smallest spatial primitive to handle and that it can be manipulated to define all geospatial data elements. It integrates methodologies from both earth sciences and computer science. Every unique identifiable spatial instance in form of point is given an identifier and it is this that is manipulated during geometry adjustment to transfer updates from the source/reference data to the data set being updated/adjusted. This approach solves the issue of silvers and dangling which are always created during data merging and the differences that are always brought into databases due to variations in data capture, storage, and manipulation approaches.

Keywords: Data Integration, Point Primitives, Geometry updating and Adjustment, and Geospatial Management

1 INTRODUCTION AND OVERVIEW

The Geo-Spatial Information Systems (GIS) are systems with hardware, software, personal to capture, store, manage, manipulate, and produce models and analyses for decision making and planning from earth reference data. GIS combines techniques from Space science, Information science, Computer science, and Management science to achieve its objectives.

Most GIS use map layers to organize the geographic features. Each layer describes a certain aspect of the modeled real world because it provides a natural technique to organize the data from possibly different sources and it can be more efficient in terms of data storage and data manipulation (Oosterom, 1994).

As the physical features and land uses of the earth constantly changing, so is the need to continuously update these layers in GIS databases. But storing map layers separately makes it impossible to directly solve topological queries that relate to features which belong to different layers (Oosterom, 1994). In order to make most of spatial decisions and solve queries that deal with multiple layers, users need to integrate geo-spatial data from different sources and a map-overlay operation has to be performed first. This is almost done by everyone as it is difficult for an individual to have all the data. This makes map-overlay one of the key operations in a GIS as it allows the user to integrate spatial data.

Before geospatial integration or merging can take place, the first step is feature matching or spatial object matching and this is achieved by algorithms based on either geometric matching, topological matching, or semantic matching (Moosavi and Alesheikh, 2006). But after that, because of the various and multi-sources of data, the different GIS data sets always can't match well. In the process, overlay operation intersect objects making area (polygon) and line features to be chopped into many pieces. This creates new layers and features in the process (Martinez et al., 2009), which are unwanted and sometimes silver polygon, dangling node and lines. Although these maybe important in case the aim is to explore the relationships between spatial, but it means the GIS database is not properly updated with the required data, if the need is to merge two data sets so that they are up to date. These new features are not needed since the aim is to update data with the changes so that new spatial databases depict what is currently on the ground. This is needed especially for decision making, service delivery, and planning. In order to update the GIS database effectively, there is always need to adjust the geometries of the objects which made up Geo-Spatial data.

However, geospatial data comprises of many complex objects with varying geometries. This makes it difficult for GIS practitioners and geospatial data managers to integrate thematically similar data sets easily in case of geometrical variation due to source, method of collection, instrument used, method of storage, and projection parameters used.

One of the reasons is lack of geometrical primitive based approach which can be used to adjust and eliminate discrepancies caused by geometry differences between objects of thematically similar data. This not withstanding the fact that, geo-spatial data can be stored in text formats which provide environments to access individual details on the object properties. One such property is the geometry which is represented as a string of coordinates with information on how to produce them graphically. In this work, we look at an algorithm which can be used to adjust mismatches in thematically similar spatial data sets basing on primary geometrical primitives - points.

By converting the different mis-matched geometries (points, polylines, and polygons) into points, then establishing the geometry differences "errors" for the identified mismatches, the GIS data object geometries can be adjusted by changing the coordinate values as per requirement for each point, then reflecting them in the GIS data sets being updated. The term adjustment is used in the context of modification, alignment, alteration, fine-tune, correction, amendment, or improvement of GIS database so that geometry of spatial features in thematically similar data sets are the same. And since in matching objects, if the level of similarity is derived from its shape, color, size, the approach is called feature based matching (Moosavi and Alesheikh, 2006). We shall expand on the same terminology and we shall have "feature adjustment" where the local adjustment will be based on shape of the geospatial object.

The approach has the advantage of avoiding the need for conversion between geodetic coordinate systems and topological relation creation and building algorithms (Li et al., 2008) as only identified mismatch geometries are obtained and adjustment applied to already existing topological data.

1.1 The Gap and Contribution to Knowledge

With the point based local adjustment approach, we are of the view that we are filling the gap and contributing to the body of knowledge in the following:-

a) The use of point-based approach to adjust and align geo-spatial data sets. GIS data is always a mixture of point, line, and polygon features/objects, so the use of points brings in a single approach to geometry adjustment and alignment.
b) The adjustment of geometry of individual features without affecting the whole GIS data. Were only the lines, points, and polygons and even sections of polygons can be adjusted and aligned to conform to the proper location which could be known ground coordinates or corrected data set.
c) The adjustment and alignment of geometry without worrying about topology and attribute loss and management of data objects since topology is never removed
d) More understanding of computational geometry in improving GIS. As (Bayer, 2008) put it, the representation of the earth using planar structures (e.g. points, lines, polygons) has its own problems that can be effectively solved using computational geometry

We extend (Martinez et al., 2009) algorithm of polygon Boolean operations and get additional support by expanding on an algorithm for polygon clipping and for determining polygon intersections and unions by (Liu et al., 2007). We use computation geometrical suggestions from (Bayer, 2008) with support from vector matching algorithm by (Moosavi and Alesheikh, 2006). We come up with an algorithm which keeps dividing segments into two parts and marking the location of division with a point which can be used in adjustment. Because it is Boolean based, it makes the algorithm fast since it with only two kinds of events: left and right endpoints of the segments (Martinez et al., 2009). Our work stops at adjusting the features, but incase there is need to rebuilt polygons and lines (Li et al., 2008), (Sester et al., 2007), (Moosavi and Alesheikh, 2006) and (Martinez et al., 2009) outlines and shows how the edges can be connected to form polygons.

1.2 The question

When geo-spatial decisions are to be made, most of the time it requires bringing together data from different sources. For example, the re-planning of a city; we use geospatial data from various department like water, power, gas, roads, building, schools, wildlife, telephone, transport, and city boundaries. These data sets have to be merged before any simulation can be carried out to aid in planning.

- During the merging of data sets, there happens to be an overlap or opening when the geometries of feature intersect. Then the question comes, what can be done to only eliminate those mismatches which are localized only to certain features in data set.
- What can be done with changes in the geometry composition because of data acquired using different models and methods, varying precision of instrument, acquired at different times and for different land use
- what about if there is need for change from point to line to polygon and vice viz.

1.2.1 Adjustment problem in perspective

Let us pick a small section of part of a street in Groningen, Netherlands; we have the cadastral plots on which residences are built, the path used by cyclist, the flower garden used to beautifully the city, and the road - see Figure 1

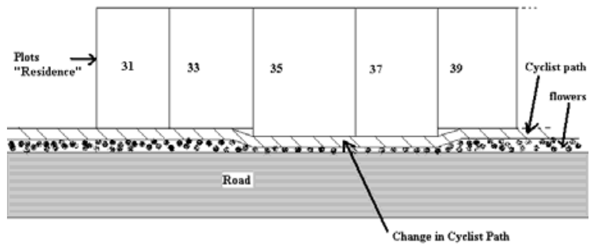

Figure 1: Geometrical Adjustment Problem

On Figure 1, we can observe the actual situation on the ground with the plot demarcations sharing the boundary with the cyclist path, which in turn share its boundary with the flower garden sharing the edge with the tarmac road. We notice that plots 35 and 37 are longer than the plots 31, 33 and 39. As a result the cyclist path is pushed outwards when it reaches plot 35 and 37 and in the process reducing on the width of the flower garden along that location. But it does not affect the road.

Going by representation on different thematic GIS layers for the roads and cadastre; we observe on Figure 2 that the road is represented as straight, but on the other hand on Figure 3; the plots actually show the difference in their sizes.

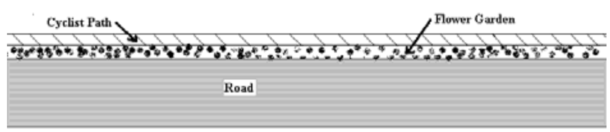

Figure 2: Road layer

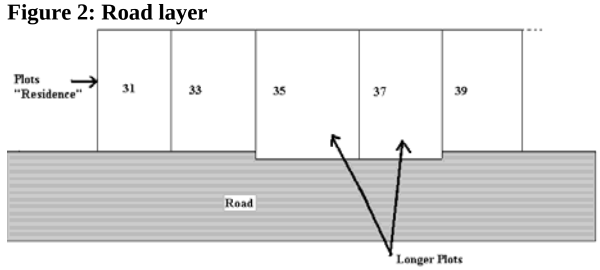

Figure 3: Cadastral Layer

Analyzing the three figures, leads us to conclude that when we merge features on Figure 2 and Figure 3 will not give us exactly Figure 1, which is the actual situation on ground.

There is need, if the two thematic GIS layers (Figure 2 and Figure 3) are to be merged, for the road layer to be adjusted at the location of plots 35 and 37. If that is not done, it will create small polygons when plots 35 and 37 intersection with the road and for proper GIS database those will have no meaning and will be miss leading. If we proceed with cleaning GIS database, it will delete those small "silver" polygons and resultant database will not depict the ground situation.

Thus the need for an algorithm which can adjust localized geometries on geospatial features, in this case adjust only the part of the cyclist path at locations of plots 35 and 37 so that the actual appearing on the ground is reflected in the database and in the merged results.

2 RELATED WORK

In this section, we described research which has and is taking place in geospatial data integration together with pro's and con's to various approaches and algorithms for map overlay, feature matching, and object adjustment and alignment in relation to geospatial data storage, management, and usage
Geospatial data is captured using different reference systems and geodetic datum like WGS-84 ellipsoid corresponding to the WGS-84 (World Geodetic System-1984) geodetic coordinate system used by GPS, Arc 1960 ellipsoid used for most cadastral work, and plane coordinate system used in GIS.

The different data sets have to be used together and managed under one system, this led to various matching algorithms finding their ways in GIS. (Moosavi and Alesheikh, 2006) gives a detailed account of why including:-

a) need to integrate scattered datasets for a new application to improve accuracy and reduce the costs of collecting data
b) multi-representation of objects to have spatial databases consist of several geometrical elements with different scales which improves automatic cartographic generalization (Bayer, 2008)
c) uniformly to present a database such that data updating can be done on other collections
d) sharing spatial data (Liu et al., 2007)
e) integration into one digital databases to reduce on data usage complication which is difficult even when data sets related to each other (Najar et al., 2006)

Different operations and many algorithms exist for carrying out clipping and finding intersection between two datasets. Some focus on merging similar geometric objects (Moosavi and Alesheikh, 2006) including exchange of attributes or for homogenizing geometry, address semantics heterogeneity of spatial data sets, improvement of quality in case one data set is captured to a higher quality, multi representation based on the matching of two datasets using Unified Modeling language (UML), selecting matching objects using Structured Query Language (SQL). But non handles point based local object adjustment.

For polygon overlays, many are explained by (Martinez et al., 2009) and (Liu et al., 2007) which can work on different polygons like convex, rectangle, and concave polygons with some requiring complex and specific data structures and others computing Boolean operations on polygons which help to determine the intersection of segments (Martinez et al., 2009). Although these operations determine the spatial coincidence (if any) of two data layers, they offer little help in updating geospatial datasets and they focus on polygon intersection when the actual fact is that geospatial data is composed of points, lines, and polygons primitives.

For the approaches which handle combined primitives, we find a huge number of research efforts in this domain:- a) approaches dealing with geospatial object matching, and b) methods for geospatial object intersection or database updating. Thus, different schemes have been proposed :- i) layer's overlying approaches which use algorithms for different geospatial data merging tasks, and ii) more specific ideas, using individual geospatial object adjustment strategies, to achieve full integrated solution. The latter is the focus of this work dealing with geometry adjustment and alignment basing on current approaches like Boolean polygon matching (Martinez et al., 2009), alignment algorithms (Tian and Kamata, 2008), computational geometry (Bayer, 2008), and object adjustment in corresponding data sets (Sester et al., 2007).

Algorithms which handle points include the plane-sweep algorithms (Oosterom, 1994) which first sort the points of line-segments based on their x-coordinate which means it does not employ an equal role for x and y values; which is needed as it could make better use of the "local processing" principle by taking both dimensions at the same time into account. But as (Oosterom, 1994) explains further that this is what happens in the uniform grid algorithm, but this algorithm does not perform well when data is non-uniform distribution and he presents the R-tree based map-overlay algorithm which tries to overcome the problems of the existing algorithms. However, it does handle the partial object or segment geometry adjustment.

For the situation which requirement to move objects using set requirements which work as constraints in the process of displacement like fixing one object and moving others then it can be achieved using the optimization technique of Least Squares Adjustment (LSA) basing on the neighborhood derivation from a Delaunay-Triangulation (Sester, 2000)

Least Squares Adjustment using Jacobean matrix of the derivations of the functions maybe good for determining the unknowns in big data sets (Sester, 2000), but does not offer solution for specific part of object to be adjusted as it gives a global solution, where a displacement of one object occurs in accordance with all its surrounding objects. However, LSA is important in our approach as change of length of the objects' sides gives an indication for their deformation and the same holds for the change of the internal angles, which helps to decide which sides or segments to adjust. Also, LSA gives a measure for the absolute positional accuracy using change in the coordinates (Sester, 2000).

For the point based feature matching presented by (Moosavi and Alesheikh, 2006) is about finding the best point to do the corresponding matching using weighted average between length and azimuth differences for control point parameter and target point and it involves normalization equation for distance parameter. This is good and can be employed in selecting the objects to use as source objects and adjustment objects in our approach.

In our approach we use the strategy of moving specific points using their x and y coordinates because they can be obtained and the adjustment should not be based on others factors like what is the best location or shape from the many as that is handled by many existing algorithms like those presented in (Moosavi and Alesheikh, 2006).

3 DESIGN

3.1 GIS Data Geometry Composition

As we design a local geometry adjustment (LGA), we have to remind ourselves that geospatial data is composed of many complex feature geometries which make it difficult for users to develop one approach to adjust the discrepancies among thematically similar data sets. Global methods of data comparison and alignment like "automatic image-map alignment problem using a similarity measure named edge-based code mutual information" by (Tian and Kamata, 2008) are good for image adjustment and use global transformation parameters which do not take care of the local "individual feature" changes. Also based on polygon clipping, intersection, or overlay (Martinez et al., 2009) and (Liu et al., 2007) are good if we do want to keep all features as these create new features and also are important in intersecting polygons and we employ them to indicate the differences in polygons on which we base to apply the LGA. But such algorithm can not be applied for situation where we need to move an edge or segment of a polygon or to reshape lines or move locations of points.

For GIS data adjustment, there are certain requirements which have to be filled

- the meaning of the shapes – keep the attributes
- maintain the relationship between features – topology
- separate data into layers for easy modeling and analysis

On that, we add the following in order to accomplish our work

- should be possible to handle individual localized primitives/parameters
- able to handle points, lines, and polygons or any combination of them

3.2 Framework for adjustment

The framework being used for the whole approach starts with two geospatial data sets (one with adjust objects which have to be adjusted and the other with source object where adjustment parameters will be computed from) and ends with adjusted data set only, see Figure 4

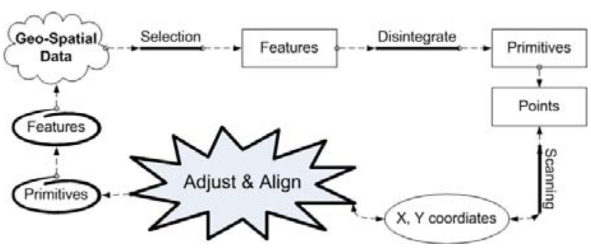

Figure 4: Framework for Geometrical adjustment

The process starts with having geo-spatial data from which we select the features which have to be adjustment. The features are then disintegrated into primitives (points, lines, and polygons) and finally into points. The coordinates of the points are then scanned and compared with the required or the proper geometrical shape and location. The process of adjustment and alignment computes the needed adjustment by finding the difference between the correct and the existing coordinates or locations. The determined adjustment values are applied to the existing coordinates/geometries and converted back into primitives, which are then combined to get the features which make up the data sets.

There is already existing algorithms in literature for the rest of the actions required apart from the local based adjustment and alignment which is the focus of this work. We are looking at adjustment of data which maybe point, line, polygon, or a certain combination.

3.3 Why point-based Approach

With various needs for object geometry adjustment in order to achieve perfect alignment of spatial objects, the point-based approach has been developed because of the advantages it offers to supplement existing adjustment algorithms, the strength include:-

- We are able to use the actual coordinates of the points, nodes, and vertices of the different features. As points, nodes, and vertices define the segments, edges, shape, size, polygon and orientation of GIS objects
- We are able to overcome the need for the global object transformation parameters as points obtained from the different GIS objects can be directly implemented in the adjustment and alignment of spatial objects.
- With points, we are able to work with local geometry transformations parameters of individual spatial objects.
- Points offer the opportunity to adjust geometry of certain features with out affecting other spatial objects in the data set.
- Points helps to just adjust part or segment of the object/feature which has the mismatch
- Using points we are able during adjustment to maintain the part being adjusted as part of the whole feature.
- We do not need to separate attribute information from the feature during adjustment
- As a compliment to existing algorithms developed, there is no need for user to think about the different shapes when implementing adjustment – the only focus is the points.
- Using points, the adjustment valves/parameters obtained can be easily handle using simple data structures
- Topology is not affected as the change takes place on the part or segment only being adjusted.

4 DESCRIPTION OF ALGORITHM

4.1 Steps for adjustment

Here we outline the steps for localized geometry primitive adjustment algorithm

- Before geospatial integration or merging can take place, the first step is feature matching or spatial object matching and this is achieved by algorithms based either on geometric matching, topological matching, or semantic matching (Moosavi and Alesheikh, 2006). In our case we based on geometric matching using geometric parameters such as lengths, azimuths, distance, and control points. With specific interest being length of objects' segment, and distance between adjustment objects. We introduce the concept of "adjustment distance" since the focus is not of matching but to update geometry of one spatial object "adjustment object" basing on the changes which can be got from another object "source object" or any other source of adjustment values like coordinates. The difference between the two objects gives us the adjustment distance/parameter. The adjustment distance is vector which is the resultant distance obtained from coordinates of the adjustment object (Xa, Ya) and the source object (Xs, Ys).
    - Selection of the needed object/feature to adjust. The selection works on two fronts – first the object "adjustment object" which need to be adjusted and the object which is providing the adjustment values "source object" or just to select the desired location or to enter the source of adjustment values. We do not need to select the whole feature, The shape can not be used as this brings in numerous errors which still later have to be eliminated. Also the shape may not be correct and

may need to be adjustment. It is not always true that if two shapes are perfectly the same, then they are correct.
  - Picking the nearest points to be used in adjustment as does not always mean that it belongs to the feature/object
- Just only the part to be adjusted.
  - We do not need to use polygon clipping, intersection, or overlay (Martinez et al., 2009) as they create new features, but we used them when selecting objects or as an indication of the differences where we have to apply the adjustment
- Identification of the input object/feature or values to use in the adjustment. Certain situation analysis were to be put into consideration when selecting values to be used in adjustment
- Actual conversion into point primitives is not done but we used points as the basis of obtaining adjustment information. This is needed as the adjustment object may not need to be wholly adjusted, but just part of it and each part may require varying adjustment. So by handling the object or segment of the object as points gives us room to move specific parts. This helps us not to focus on how to detach the corresponding properties and also to maintain the original data sets like colors and representation styles.
  - After picking the segments or objects then handle them as points. We do not break features into primitives but use points as the basis for obtaining adjustment values. This point based adjustment data have to be kept as a series of points in a certain order. Thus the points from two sources of values (say two objects) for adjustment should be stored in the same direction to help in reference and reading corresponding values for adjustment.
  - When objects are handled as points, they have to be stored and labeled in same way in the corresponding data sets
  - The interval for point selection and creation is based on the scale in the GIS database.
  - The two sources of data have to be handled as points at the same scale. When ever there are differences in scale, it has to be made the same so that the final picking of segments for conversion into points is done at the matching scale.
  - For polygons, we use the surface/edges in form of segments instead of the polygon's gravity center (Moosavi and Alesheikh, 2006)

- Because of localized need for adjustment and alignment, the global transformation (translation, scaling, rotation) are not applicable during computation of adjustment values.. We use the local transformations which are able to move each point or segment individually independent of the neighbors.
- There are certain considerations we consider on deciding on how to handle adjustment – if the difference is positive or negative, just add to the selected object values. No need to worry if the difference is positive or negative, whether the adjust is to handle gaps or openings
- When we were carrying out adjustment, we had to make some hard decisions especially on components of the adjustment algorithms
  - How to handle point to point alignment
  - Point existing alone when it should be part of a line
  - Point should have been part of the polygon
  - Point located inside a polygon
  - Point located along a line
  - Line to line alignment
  - Line but should be a point
  - Independent line but should be part of a polygon
  - Line along polygon which should be part of
  - Polygon should be point
  - Polygon should be a line
  - Polygon should be combination of points and lines
  - Polygon to polygon alignment

- In case, there is need to build features from the primitives; we are not adding any new to already existing algorithms for building features/objects and topology in case there is need.
- For deleting, inserting, creating new features/objects there are already algorithms and those are well handle in many GIS

4.2 Getting the adjustment points

During adjustment, we select the whole part of the feature to be adjusted and corresponding part from the source object. If the selected parts or segment are not the same length, the longest is used as the basis for getting adjustment values along points. This is done so as we do not want to loss any details – so it is assumed that the longest segment has more details but even if it is not the case, all the details for the shorter segment are always picked. The longer segment is divided into the required intervals using the set length T and the short segment will be also divided into a certain number of intervals using the length T. Then as adjustment starts, the short segment has to enlarged to fit the longer segment. The process continues by dividing into smaller intervals of length T whenever the previous segments double the length T. The points at the end of the segments (vertices) in the two segments will be matched to generate adjustment values Xa, Ya, Xs, and Ys
where

- Xa and Ya are the coordinates on feature to be adjusted,
- Xs and Ys are the coordinates for source object which should be the proper location of the selected object – these could be values obtained from the geometry or values from a known correct coordinate reading.

We introduce a vertical line (perpendicular to the segment) as used by (Martinez et al., 2009) where by we move the vertical line at calculated intervals (set by user in relation to scale) and at every moment as the segment intersects with the vertical line, the coordinates are read off then ordered by x coordinates Figure 5. They are given an incremental identifier and stored in a data structure.

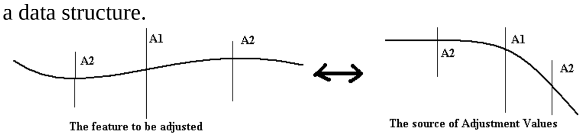

Figure 5: Process of getting Adjustment Values

It starts off with level A1 which divides the segments into two parts, followed by the next level A2 which further divides the segments obtained at level A1 into two. The process continues until we reach the required level according the data scale.

4.3 Computation of Adjustment Transformation Values

To compute the adjustment values, we use
dx = (Xs – Xa) and dy = (Ys – Ya)
where

dx and dy are the differences in X and Y readings, S_{12} is the distance between adjustment object and the source object

Since all GIS has x and y coordinates, we take advantage of these values and we do determine the difference change individually in x and y and those are applied to the object/feature directly to move it or adjust its geometry.

Also the object orientation (arctan dx/dy) which is the direction of one object side in relation to others is not considered as the whole adjustment object is not always handled or taken as a whole. We do not handle angle of rotation as we are only dealing with a two dimensional geometry.

We should note that we do not rebuild relation and topology as the original figure/features are disintegrated. The other advantage is that we do not need to worry about adjusting the neighboring features as it is automatically done as the adjustment takes place when topology is present.

4.4 Computation of Adjustment Values

To compute the adjustment values, we use the formula
$Xp = Xa + dx$ and $Yp = Ya + dy$
where

> Xp and Yp are the coordinates of the final adjusted object/feature. We need to compute these as sometime the Xa and Ya cannot be obtained on the map or as input from a third source of information. If true values for X and Y can be obtained, then dx and dy will be zero

In this algorithm, we use the actual x and y reading as it eliminates the unnecessary computation which would involve comparing and determination the resultant adjustment values for the movement of the object geometry which also requires the direction.

4.5 Updating Adjustment Object

With the identified points and segment for updating, obtained adjustment parameters, and known coordinates for the adjustment object (Xa, Ya); we proceed to update the adjustment values

To test the geometry update algorithm, we used the plug-in approach in Quantum GIS (QGIS) developed by Qt C++ framework.

Qt supports the development of cross platform GUI applications with its "write once, compile anywhere" approach. Using a single source tree and a simple recompilation, applications can be written for Windows, Mac OS, Linux, Solaris, and many other versions of OS.

Plug-ins are a great way of extending the functionality of QGIS without changing the QGIS internals. To compile the code on our computer we needed: the gnu-compiler g++, make, qt4, the qt4 development tools qmake, moc, uic, and the qgis libraries and header files.

5 FUTURE WORK

We are moving towards testing the geometry updating in PostGIS so that we move in-line with the international approach of using simple feature specification. We also want to test the implementation of our geometry updating and adjustment using python to get a plug-in for Quantum GIS so that we can conclude on the robustness and computational performance.

6 CONCLUSION

In this paper, we have shown that geospatial data integration can be effectively carried by incorporating geometry adjusting to update one data set with changes or information from another data set or source. This can be easily done by using points as the simplest and smallest spatial primitive that can be manipulated to define all geospatial data elements in that every unique spatial instance is identified in form of point and is given an identifier and it is this that is manipulated during geometry adjustment to transfer updates from the source data to the data set being updated. With this we obtain a uniform geometrical updating, which flows into effective integration of data from a variety of sources that is essential for increased understanding and decision-making about the earth through answering complex questions in geospatial information systems.

REFERENCE:

BAYER, T. (2008) The importance of computational geometry for digital cartography. *Geoinformatics FCE CTU.*

HILD, H. (2001) Automatic Image-To-Map-Registration of Remote Sensing Data. *Photogrammetrische Woche 2001, Fritsch/Spiller (Eds.).* German.

LI, C., LIU, Y., WANG, Q. & TAI, Y. (2008) A GIS Database Updating Algorithm Based on GPS Surveying Data. *2008 International Conference on Computer Science and Software Engineering.*

LIU, Y. K., WANGA, X. Q., BAO, S. Z., GOMBOSI, M. & ZALIK, B. (2007) An algorithm for polygon clipping, and for determining polygon intersections and unions. *Computers & Geosciences,* 33**,** 589-598.

MARTINEZ, F., RUEDA, A. J. & FEITO, F. R. (2009) A new algorithm for computing Boolean operations on polygons. *Computers & Geosciences,* 35**,** 1177–1185.

MOOSAVI, A. & ALESHEIKH, A. A. (2006) Developing of Vector Matching Algorithm Considering Topologic Relations. *MME08 PN 40.*

NAJAR, C., RAJABIFARD, A., WILLIAMSON, I. & GIGER, C. (2006) A Framework for Comparing Spatial Data Infrastructures An Australian-Swiss Case Study. *GSDI-9 Conference Proceedings.* Santiago, Chile.

OOSTEROM, P. V. (1994) AN R-TREE BASED MAP-OVERLAY ALGORITHM. *EGIS Foundation*.

SESTER, M. (2000) GENERALIZATION BASED ON LEAST SQUARES ADJUSTMENT. *International Archives of Photogrammetry and Remote Sensing.* Amsterdam.

SESTER, M., GÖSSELN, G. V. & KIELER, B. (2007) Identification and adjustment of corresponding objects in data sets of different origin. *10th AGILE International Conference on Geographic Information Science 2007.* Aalborg University, Denmark.

TIAN, L. & KAMATA, S. (2008) AN AUTOMATIC IMAGE-MAP ALIGNMENT ALGORITHM BASED ON MUTUAL INFORMATION AND HILBERT SCAN. *16th European Signal Processing Conference (EUSIPCO-2008).* Switzerland.