On the use of computer vision for quality assessment of vector data

S. Gautama & A. Borghgraef
Dept. Telecommunication and Information Processing, Ghent University, Belgium

Keywords: object recognition, road detection, change detection, GIS

ABSTRACT: In this paper we propose a system for automated quality assessment of road vector data based on very-high-resolution satellite images. The system is based on feature based spatial registration, where detected features in the image are registered to corresponding features in the vector data. For our application, crossroads were found to be stable features. They are automatically detected in the image using an improved ridge detector. Given these features, the problem of comparing image and vector data is represented as two sets of points which need to be matched. In this matching process, we need to take into account noise on the position and possible spurious points. The problem is solved using a graph matching process, where an attributed graph is constructed for each set of points using the spatial relations between points as attributes. Measures of change are proposed using the result of the graph matching process.

1 INTRODUCTION

A major challenge in the production and use of geographic information is assessment and control of the quality of the database. The rapid growing number of sources of geospatial data, ranging from high-resolution satellite and airborne sensors, GPS, and derivative geospatial products, pose severe problems for integrating data. One of the major challenges for content providers to face is the problem of upgrading their current databases to a higher accuracy and ensuring the quality of the information. Current techniques can’t support this in a cost-effective way due to the necessary manpower. Automated detection of change and anomalies in the existing databases using very-high-resolution (VHR) satellite images can form an essential tool to support quality control and maintenance of spatial information. In this paper we examine to what extent remote sensing and computer vision can be used to detect deviation of a GIS database versus the “real world”.

The system is based on feature based spatial registration, where detected features in the image are registered to corresponding features in the vector data. The system consists out of two stages: 1) a low-level feature detection process, which extracts roads and crossroads using an improved ridge detector, and 2) a high-level matching process, which uses graph matching to find correspondences between the detected image information and the road vector data. The graph matching process, based on continuous relaxation labeling, is driven by the spatial relations between the features and takes into account different errors that can occur (e.g. spatial inaccuracy, data inconsistencies between image and vector data). The matched features can be used to calculate a rubbersheeting transformation between image and vector data, using triangulation. Such a transformation is able to compensate for the local distortions that can occur between the datasets.

The paper is organized as follows. Section 2 briefly discusses the state-of-the-art in feature based registration. Section 3 introduces road detection based on the differential ridge detector. Section 4 shows the improvement that is made to the initial detection to be able to detect crossroads in the image. Section 5 shows how corresponding objects between the image and the vector data are found using graph matching. Section 6 shows how the result of the matching process can be used to define measures of change between image and vector data.

2 FEATURE BASED REGISTRATION

Feature based registration is the primary approach to compare data sources that have differing types of representation as is the case for images compared to vector data. Traditional pixel based correspondence techniques are inadequate to solve this registration
problem [1]. An additional processing step is necessary to convert the raw image information to a representation, which is closer to the vector data. This is done by detection of image features like lines, corners and segments. Given these features an abstract representation can be built as an attributed graph [2]. The graph nodes represent image features and the node attributes can contain measurements on these features. The graph arcs represent relations between features and the arc attributes can contain measurements on spatial relations. A similar graph can be build on the vector data, using vectors as nodes and relations between vectors as arcs. The problem of registration is represented as a graph matching problem, which seeks the correspondence of similar nodes between two attributed graphs.

In computer vision error tolerant graph matching techniques form an important class of techniques. These techniques seek a graph or subgraph morphism, which allows for distortions. A general distortion model defines the deletion and addition of graph nodes and arcs, and replacement of attribute values. A similarity measure or distance function between two attributed graphs is used that models the occurring distortions using heuristics. Early techniques were a generalization of string matching. More recent models are based on information theoretic principles [3] or Bayesian modeling [4]. Solving the correspondence problem is difficult and several optimal and approximate techniques have been proposed. These include among others search trees, dynamic programming, annealing and genetic algorithms. In this study we examine relaxation labeling, a popular approximate technique which has low, polynomial time complexity.

3 DETECTION OF ROADS

3.1 Ridge detection

Line structures in an image can be seen as ridges or valleys in the pseudo-terrain representation of the image, where the intensity in each pixel is interpreted as a height. A standard model to detect these lines is to calculate first and second order partial derivatives in a pixel. Pixels belonging to the centre of a line will show a low gradient, a high curvature in the direction perpendicular and a low curvature in the direction parallel to the line. The calculation of the partial derivatives can be done in various ways. The facet method [5] approximates the intensity values \( I \) in a square window of size \( N = w^2 \), with odd window size \( w \), with a polynome \( F \). The origin is chosen in the central pixel of the window. The value of the polynome \( F \) in pixel \((i,j)\) is given by:

\[
F(i, j, \hat{\theta}) = a_i + a_j + a_{ij} + a_{i^2} + a_{j^2} + a_{ij^2} + a_{i^2j} + a_{ij^2} = m^T \hat{\theta}
\]

\[
m = \begin{bmatrix} 1 & i & j & i^2 & ij & j^2 \end{bmatrix}^T
\]

\[
\hat{\theta} = [a_i \ldots a_{ij^2}]^T
\]

(1)

The facet model for ridge detection searches the least-squares solution \( \hat{\theta} \), given the image data \( \hat{x} \) containing the intensity value \( I(i, j) \) in each pixel \((i, j)\):

\[
\arg \min \ r(\hat{\theta}) \text{ with } r(\hat{\theta}) = \| M \hat{\theta} - \hat{x} \| _2
\]

\[
M = \begin{bmatrix}
1 & i_1 & j_1 & i_1^2 & i_1j_1 & j_1^2 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
1 & i_N & j_N & i_N^2 & i_Nj_N & j_N^2
\end{bmatrix} \in \mathbb{Z}^{N \times 6}
\]

\[
\hat{x} = [I(i_1, j_1) \ldots I(i_N, j_N)]^T \in \mathbb{R}^{N \times 1}
\]

This leads to the linear system \( M^T M \hat{\theta} = M^T \hat{x} \) with the solution \( \hat{\theta} \), given by \( \hat{\theta} = (M^T M)^{-1} M^T \hat{x} \). The matrix \( M \) is independent of the position of the window within the image, meaning that the calculation of \( (M^T M)^{-1} M^T \) needs to be performed only once for the processing of an image with a fixed window size \( w \). On the basis of the parameters \( \hat{\theta} \) of the interpolated surface \( F \), the gradient and Hessian in a certain pixel can be calculated:

\[
\text{gradient}(I) = \begin{bmatrix} \nabla x^2 I \nabla y I \end{bmatrix} = \begin{bmatrix} a_i + 2a_{ij} + a_{j} \\
2a_{ij} + a_j + 2a_{i} \\
a_{ij} + 2a_{i} \\
a_{i} + 2a_{j} \end{bmatrix}
\]

(3)

\[
\text{Hessian}(I) = \begin{bmatrix} \nabla x^2 I & \nabla x \nabla y I \nabla y^2 I & \nabla y I \end{bmatrix} = \begin{bmatrix} 2a_i & a_j \\
a_{ij} & a_i \\
a_{ij} & 2a_j \end{bmatrix}
\]

In first instance, we are only interested in the gradient and the Hessian in the central pixel of the window (i.e. \( i=j=0 \)). The eigenvalues of the Hessian give the principle curvatures along the principle directions of the observed image structure in each pixel. The largest eigenvalue (in absolute value) is called the maximal curvature, the smallest is called the minimal curvature. To classify an image structure as a line, thresholds need to be defined on both curvature values and gradient magnitude. A line is characterized by a large maximal curvature and small gradient magnitude and minimal curvature. The direction of the line is given by the orientation of the eigenvector belonging to the minimal curvature.

3.2 Vectorization

Given a window size and a set of threshold values line pixels can be detected in the image. The threshold operation however does not necessarily select a
pixel-width road but more likely will select a thick line of several pixels wide. To vectorize this line to a pixel chain, a morphological thinning operation is performed [6], after which the orientation information of the maximal curvature is used to link the connecting pixels which show a similar orientation.

Fig. 1 shows the result that is obtained by this process. Fig. 1a shows an extract of a panchromatic IKONOS image above a suburban area in the vicinity of Ghent. Fig. 1b shows the pixels that have been selected as ridge pixels. A window size of 11 pixels was used. After vectorization, this binary image is transformed into a representation consisting out of pixel chains. Fig. 1c shows the result after filtering out the chains with a small length. Length proves to be a very powerful feature to distinguish road structures from clutter. Fragmentation however limits the minimal length which can be imposed.

4 DETECTION OF CROSSROADS

The road network that can be detected using the previous scheme is not of sufficient quality to be useful for comparison with a road database. The main difficulty is the difference in representation between the pixel chains that can be detected from the image and the polyline vectors that represent roads in the database. This hinders the correspondence problem considerably. A much more robust registration object is necessary. We found crossroads to be good candidates since in their abstract form, they can be represented as point objects both in the image as well as in the database.

We model a crossroad as a point where lines meet. This means that built upon the road network that is detected using line detection, we look for pixels in the pixel chains which have three or more neighbors. We choose this strategy above corner detection, because corner detection gives many spurious responses not belonging to crossroads that are not easy to filter out. Our method is stays closer to the road network logic. A major problem however is that the detection of the road network fails in the vicinity of crossroads since the line model does not hold anymore. At the junction, the intensity surface will not be modeled as a valley or a ridge but as a flat spot. As a consequence, the road network will often be broken around crossroads.

We implemented a simple region growing scheme [6] which extends the initial road fragments with regions which show a similar gray value. For roads which are adequately detected, this proves to be sufficient in many cases to bridge the bad spots at crossroads. Fig. 2 shows the result after region growing. The top images show the initially detected pixel chains and the associated pixel regions with a width of 3 pixels. The bottom images show the resulting regions after region growing and the pixel chains after vectorization.

The simple scheme is of course not foolproof. A cheap and efficient verification to filter out false alarms is to check if in the vicinity of a hypothetical crossroad a flat spot exists. Crossroads are seen by the line detector as a flat region. By simply applying different thresholds on the gradient and the curvature, which have already been calculated for the detection of lines, we can detect the flat spots and verify our crossroads. Fig. 3 shows an example of crossroad detection. The boxes are detected crossroads using the neighbor model. Crosses are crossroads which are verified and retained using the flat spot check.

5 FINDING CORRESPONDING OBJECTS USING GRAPH MATCHING

The problem of road detection is a difficult one due to occlusion, shadow and variability in shape and radiometric properties. This is especially true for urban zones where these problems are more pronounced than in rural areas. Due to the complexity of the problem, detection of roads cannot be performed in a single pass. The road fragments that are extracted in early vision need to be grouped into a
consistent road network using “global” information, in addition to the “local” generic road model that is used.

Global information can come from several sources:
- road network logic (e.g. cars on the roads, houses next to roads, no gaps);
- an existing road network that needs to be updated or checked;
- additional GIS layers that add constraints (e.g. parks, houses)
- user interaction

Part of the road network logic was already used in building up the initial detection (e.g. the line structure, flat spots for crossroads, no gaps, similar grey values etc.). The second source of information that is exploited is the existing road network in the database. We will not yet look at improving our detected road network but try to find a spatial registration between the image and the database. In the previous section, crossroads were found to be stable registration objects. The problem can then be represented as finding the correspondence between two sets of points: one originating from the image and one which can be extracted from the database. This can of course be applied to situations other than image-to-GIS registration (e.g. image-to-image or GIS-to-GIS).

5.1 Continuous relaxation labeling

The matching problem can be defined as a graph labeling problem. The following are defined:
1. a set of objects \( i \), corresponding to image features;
2. a set of labels \( \lambda_i \), corresponding to GIS features;
3. a neighbour relationship over the objects;
4. restrictions on possible labels between pairs of neighbouring objects.

Relaxation labeling techniques use an iterative process to determine the probabilities of each object. Different update rules have been proposed. In [7] the relation between different update rules is analytically shown. The problem of finding consistent solutions is shown to be equivalent to solving a variational inequality which is based on the mathematical concept of “consistency”. This concept is interesting because it lays bare the foundations of the labeling process and offers guidance in determining good compatibility coefficients.

To each object \( i \) a probability distribution is associated that \( i \) has label \( \lambda_k \), \( p_i(\lambda_k) \): 
\[
0 \leq p_i(\lambda_k) \leq 1, \quad \sum_{\lambda_k} p_i(\lambda_k) = 1 \quad (4)
\]

For each pair of neighbouring objects \( i, j \) and for each pair of labels \( \lambda_k, \lambda_l \), a compatibility coefficient \( r_{ij}(\lambda_k, \lambda_l) \) exists. These coefficients express the compatibility of assigning label \( \lambda_k \) to object \( i \) in combination with assigning label \( \lambda_l \) to object \( j \). Negative values express incompatibility, positive values compatibility. Given these quantities, the support of a label \( \lambda_k \) for the object \( i \) given by the correspondence \( \bar{p} \) is defined by
\[
s_i(\lambda_k, \bar{p}) = s_i(\lambda_k, \lambda_{\bar{p}}) = \sum_{\lambda_l} \sum_{j=1}^{N} r_{ij}(\lambda_k, \lambda_l) p_j(\lambda_l) \quad (5)
\]

To find a consistent labeling, we optimise the average local consistency, given by
\[
A(\bar{p}) = \sum_{i=1}^{N} \sum_{k=1}^{\lambda_i} p_i(\lambda_k) s_i(\lambda_k, \bar{p}) \quad (6)
\]

which is a quadratic function which we optimise using a constrained gradient descent method, taking into account the restrictions of Eq.(1).
5.2 Matching sets of points

The technique makes use of the spatial relations between points to find correspondences. This makes the correspondence technique less vulnerable for road junctions which are detected in the right location but whose detected shape does not fully correspond to its counterpart in the database (defects can occur due to image noise). So instead of using features based on the shape of a single crossroad to guide matching, we use geometric relations (e.g. angle, distance) between a crossroad and its neighbours to find correspondences. These are much more stable features given the detection quality which we can realistically expect from line detection. In these experiments we rely only on the relative angle between pairs of points. Fig.4 shows an illustration of this constraint. The black points show object points and the grey points show the label points. In mapping a pair of points and the relative angle between the lines and should lie within a margin (e.g. ±π/4). Although this is a weak constraint, it has been shown that the constraint is adequate to find correspondences between sets of points, disturbed by location noise [8].

5.3 Experimental results

For our experiments we investigate the automatic registration of the IKONOS image of Ghent with a road vector database. The image is a standard panchromatic GEO product. An extract of the region of the vector data is shown in Fig.5. This figure also shows examples of the IKONOS image in overlay and illustrates some of the problems that can occur. Although the observed terrain is flat, relatively large inconsistencies exist between the two data sources. In addition the database that was used, did not contain an explicit crossroad class. We have taken the feature points describing the break points of the vector polylines as crossroads. For the region under examination this approximates the crossroads in the scene since the region contains mainly straight streets. A number of spurious points is falsely introduced in the vector set of points, due to points which do not belong to crossroads (e.g. points defining a roundabout). These points however were left in the dataset as additional source of noisy structures.

Using the detected crossroads in the image and the set of feature points from the vector data, relaxation labeling was applied to find the corresponding crossroads in the two datasets. The image dataset contains 82 points, the vector dataset contains 205 points. The correspondent of an image point is searched within a radius of 150m around the position of the image point in the vector dataset. In our case, this amounts to an ambiguity of around 10 candidate vector points per image point. By performing a manual registration of the image points with the corresponding vector points, we calculated an RMS of 7m with a maximum error of 50m. After relaxation labeling, we obtain 71 correct correspondences, 11 null correspondences and no false correspondences.

Based on the matched points, a rubbersheeting transformation can be performed. This transformation performs a triangulation of the image set of points and calculates a local affine transformation of each image triangle to the corresponding triangle in the vector data. In this way, local distortions can be compensated, which is not possible if a global transformation would have been used. Fig.6 shows the result of the registration process. Of course, for this type of local transformation it is important that there are very few false correspondences. Outliers could however be detected by comparing the local affine

---

Figure 4. Relational constraint between pairs of points

Figure 5. Road vector data and some typical inconsistencies that can occur.
parameters of each triangle after registration and filtering out extremal values.

6 COMPARING DATA USING MATCHED OBJECTS

As was shown in the previous section, the registration process is able to compensate local distortions. A second step is to see how the information of matched crossroads can be used to detect change in the vector data. Since we know the local transformation parameters of each triangle, change between the datasets can be measured based on this. Fig. 7 shows an illustration of this principle. An artificial problem is shown, containing two sets of points: one set of randomly scattered points and a second set which is a copy of the first set but with added noise on the spatial position. The first quadrant of the image contains increased spatial noise compared to the other quadrants. The matching algorithm in this particular case finds all correct correspondences. Using the matched points, the triangulation of the first set of points, marked in gray in Fig. 7(a), can be compared to the triangles that are obtained using the matched points in the second dataset, marked in black. Note that the triangles in the second dataset do not necessarily form an exact triangulation of the set of points due to the spatial noise that was introduced. Fig. 7(b) shows detected regions of change by comparing the change in surface of corresponding triangles. This is a simple illustration of change detection using the graph matching result. Other more refined measures of change are possible, e.g. by directly comparing the local affine parameters compared to mean values over a given region.

7 CONCLUSIONS

We have proposed a system for automated quality assessment of road vector data based on very-high-resolution satellite images. The system is uses feature based spatial registration, where detected features in the image are registered to corresponding features in the vector data. For our application, crossroads were found to be stable features. They are automatically detected in the image using an improved differential ridge detector. The improvement is based on a simple region growing scheme which bridges the gap caused by the deviation of the ridge model in the vicinity of crossroads.

Based on the detection of crossroads, a correspondence between the image and the database was found using graph matching techniques. In this matching process, we need to take into account noise on the position and possible spurious points. The main factor here is a good definition of the constraints which define the desired solution. A good
definition of constraints allows for a good performance of the system. The problem is solved using a graph matching process, where an attributed graph is constructed for each set of points using the spatial relations between points as attributes. After matching a rubbersheeting transformation based on triangulation can be calculated, where each point has a counterpart and local affine transformation of image triangles can be performed. This is very flexible and allows for local deformations to be compensated.

Based on the matching result, measures of change are proposed using the result of the graph matching process. These allow to identify regions of change based on the change in shape of the corresponding triangles.

ACKNOWLEDGEMENTS

The authors wish to thank the support of the OSTC which has funded this research under the STEREO program. Special thanks go to Tele Atlas for the use of their data in our research.

REFERENCES