

Linear Feature Extraction Using Perceptual Grouping And Graph-Cuts

Charalambos Poullis
CGIT/IMSC, USC
Los Angeles, CA 90089
charalambos@poullis.org

Suya You
CGIT/IMSC, USC
Los Angeles, CA 90089
suyay@graphics.usc.edu

Ulrich Neumann
CGIT/IMSC, USC
Los Angeles, CA 90089
neumann@graphics.usc.edu

ABSTRACT

In this paper we present a novel system for the detection and extraction of road map information from high-resolution satellite imagery.

Uniquely, the proposed system is an integrated solution that merges the power of perceptual grouping theory (gabor filtering, tensor voting) and segmentation (graph-cuts) into a unified framework to address the problems of road feature detection and classification. Local orientation information is derived using a bank of gabor filters and is refined using tensor voting. A segmentation method based on global optimization by graph-cuts is developed for segmenting foreground (road pixels) and background objects while preserving oriented boundaries. Road centerlines are detected using pairs of gaussian-based filters and road network vector maps are finally extracted using a tracking algorithm.

The proposed system works with a single or multiple images, and any available elevation information. User interaction is limited and is performed at the beginning of the system execution. User intervention is allowed at any stage of the process to refine or edit the automatically generated results.

1. INTRODUCTION

Traditional road mapping from aerial and satellite imagery is an expensive, time-consuming and labor-intensive process. Human operators are required at every part of the process to mark the road map in the images. In recent research, several computer vision based systems were proposed for the semi-automated or automated road map extraction. However, the gap between the state of the art and the goal still remains wide. Currently, there is no existing method which allows for the complete and reliable extraction of road map information from aerial and satellite imagery. Hence, there is still a need for an automated procedure that can reliably and completely detect and extract road networks and update road databases in Geospatial Information Systems.

In this work, we propose an integrated system for the automatic detection and extraction of road networks from aerial

and satellite images. To our best knowledge, there is no work done in combining perceptual grouping theories and segmentation using graph-cuts for the automatic extraction of road feature.

Gabor filters tuned at varying orientations and frequencies are used to extract features of special interest. A tensorial representation which allows for the encoding of multiple levels of structure information (points, curves, surfaces) for each point is used to represent the features extracted by the filters. Such a representation is very efficient when dealing with noisy, incomplete and complicated scenes. The classification of the encoded features is then based on a tensor voting communication process that is governed by a perceptual field, encoding the constraints and rules of how a point receives/casts votes from/to its neighbors. The accumulation of votes at each point provides an accurate estimate of the features going through the point (Section 3).

An orientation-based segmentation using graph-cuts is used to segment the road candidates using the refined feature information resulting from the tensor voting process (Section 4). The road centerlines are then detected using a pair of bi-modal and single mode gaussian-based kernels which respond to parallel lines and flat areas respectively. Finally, a tracking algorithm is used to extract the road network vector map (Section 5).

2. BACKGROUND AND RELATED WORK

Several techniques have been proposed and developed so far and can be separated into three main categories: pixel-based, region-based and knowledge-based.

2.1 Pixel-based

Edge detection. The goal of edge detection is to find at which points in the image the intensity value changes sharply. Although, many edge detectors [4, 15, 14, 6] already exist, none of them can extract complete road segments from a given image. Instead, the output is a list of possible edge points which have to be processed further. In [1] lines are extracted in an image with reduced resolution as well as roadside edges in the original high resolution image. Similarly, [11] uses a line detector to extract lines from multiple scales of the original data. [17] applies Steger's differential geometry approach [16] for the line extraction.

Road detection. Road detection is performed using a model of a road or a modified edge detector. The goal is to detect the roads as parallel lines separated by a constant

width. In [10] they use a multi-scale ridge detector [16] for the detection of lines at a coarser scale, and then use a local edge detector [4] at a finer scale for the extraction of parallel edges. Linking the two edges together, creates a "ribbon", which is then optimized using a variation of the active contour models technique-snakes introduced by [9].

2.2 Region-based

The goal of the region-based techniques is to segment the image into clusters using classification or region-growing algorithms. In [19] they use predefined membership functions for road surfaces (spectral signature, reflectance properties) as a measure for the image segmentation and clustering. Likewise, in [5] they use the reflectance properties, from the ALS data and perform a region growing algorithm to detect the roads. [8] uses a hierarchical network to classify and segment the objects.

2.3 Knowledge-based

Knowledge can have several forms. In [17], human input is used to guide a system in the extraction of context objects with associated confidence measures. A similar approach is used for context regions where rural areas are extracted from the SAR images and a weight is assigned to each region. The system in [18] integrates knowledge processing of color image data and information from digital geographic databases, extracts and fuses multiple object cues, thus takes into account context information, employs existing knowledge, rules and models, and treats each road subclass accordingly. [5] uses a rule-based algorithm for the detection of buildings at a first stage and then at a second stage the reflectance properties of the road. Explicit knowledge about geometric and radiometric properties of roads is used in [17] to construct road segments from the hypotheses of roadsides.

3. PERCEPTUAL GROUPING

3.1 Gabor Filtering

We employ a bank of gabor filters tuned at 8 different orientations θ linearly varying from $0 \leq \theta < \pi$, and at 5 different high-frequencies(per orientation) for multi-scale analysis. The remaining parameters of the filters are computed as functions of the orientation and frequency parameters as in [7]. The filtered images are then grouped according to the orientation of the filters, thus resulting in 8 images; one per orientation. The higher the response of a point in an image, the more likely that the feature at that point is aligned to the direction of the filter used to produce the image and vice-versa. Finally, all pixels in the resulting 8 images are encoded into tensors as explained in the next section.

3.2 Tensor Voting Overview

Tensor voting is a perceptual grouping and segmentation framework introduced by [13]. The data representation is based on tensor calculus and the communication of the data is performed using linear tensor voting.

In this framework, a point $x \in \mathbb{R}^3$ is encoded as a second order symmetric tensor which captures the geometric information for multiple feature types(junction, curve, surface) and a saliency, or likelihood, associated with each feature

type passing from that point x . A tensor T is defined as,

$$T = \begin{bmatrix} \vec{e}_1 & \vec{e}_2 & \vec{e}_3 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} \vec{e}_1^T \\ \vec{e}_2^T \\ \vec{e}_3^T \end{bmatrix} \quad (1)$$

$$T = \lambda_1 \vec{e}_1 \vec{e}_1^T + \lambda_2 \vec{e}_2 \vec{e}_2^T + \lambda_3 \vec{e}_3 \vec{e}_3^T \quad (2)$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ are eigenvalues, and $\vec{e}_1, \vec{e}_2, \vec{e}_3$ are the eigenvectors corresponding to $\lambda_1, \lambda_2, \lambda_3$ respectively. By applying the spectrum theorem, the tensor T in equation 2 can be expressed as a linear combination of three basis tensors(ball, plate and stick) as in equation 3.

$$T = (\lambda_1 - \lambda_2) \vec{e}_1 \vec{e}_1^T + (\lambda_2 - \lambda_3) (\vec{e}_1 \vec{e}_1^T + \vec{e}_2 \vec{e}_2^T) + \lambda_3 (\vec{e}_1 \vec{e}_1^T + \vec{e}_2 \vec{e}_2^T + \vec{e}_3 \vec{e}_3^T) \quad (3)$$

In equation 3, $(\vec{e}_1 \vec{e}_1^T)$ describes a stick(surface) with associated saliency $(\lambda_1 - \lambda_2)$ and normal orientation $\vec{e}_1$, $(\vec{e}_1 \vec{e}_1^T + \vec{e}_2 \vec{e}_2^T)$ describes a plate(curve) with associated saliency $(\lambda_2 - \lambda_3)$ and tangent orientation $\vec{e}_3$, and $(\vec{e}_1 \vec{e}_1^T + \vec{e}_2 \vec{e}_2^T + \vec{e}_3 \vec{e}_3^T)$ describes a ball(junction) with associated saliency λ_3 and no orientation. The geometrical interpretation of tensor decomposition is shown in Figure 1(a). Every point in the images

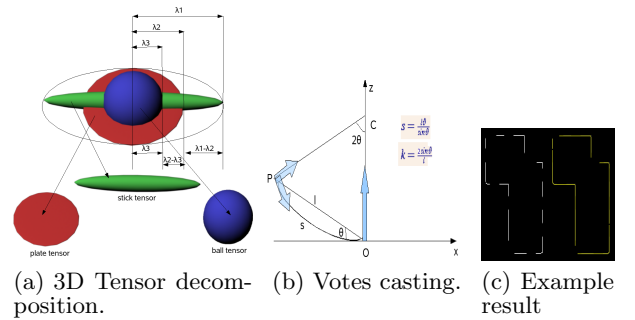


Figure 1: (a)Tensor decomposition,(b)vote casting,(c)Before (left) and after (right) the tensor voting..

computed previously is encoded using equation 1 into a unit plate tensor with its orientation $\vec{e}_3$ aligned to the filter's orientation and is scaled by its response in the filtered image, therefore giving 8 differently oriented tensors per point. Adding the 8 tensors of each point gives a tensor describing the feature types passing through that point.

The tensor encoded points then cast a vote to their neighbouring points which lie inside their voting fields, thus propagating and refining the information they carry(Figure 1(b)). For a comprehensive analysis and further details refer to [13]. Figure 1(c) shows the test image with an incomplete polygon in white and the resulting curves being overlaid in yellow. As it is obvious, most of the discontinuities were successfully and accurately recovered by this process.

A process which combines the local precision of the gabor filters with the global context of tensor voting is performed, as a series of sparse and dense votings similarly to [12]. Finally, by analyzing the information encoded in the tensors the most dominant feature type of each point is determined as the feature type with the highest associated likelihood(i.e.

$\lambda_1 - \lambda_2, \lambda_2 - \lambda_3, \lambda_3$ in equation 3). The classified curve points with their associated orientation information computed are then used to guide the orientation-based segmentation process.

4. ORIENTATION-BASED SEGMENTATION USING GRAPH-CUTS

The result is a segmented foreground image consisting of road candidate points, and a background image consisting of non-road and vegetation points.

4.1 Graph-cut Overview

In [3, 2] the authors interpret image segmentation as a graph partition problem. Given an input image I , an undirected graph $G = \langle V, E \rangle$ is created where each vertex $v_i \in V$ corresponds to a pixel $p_i \in I$ and each undirected edge $e_{i,j} \in E$ represents a link between neighbouring pixels $p_i, p_j \in I$. In addition, two distinguished vertices called *terminals* V_s, V_t , are added to the graph G . An additional edge is also created connecting every pixel $p_i \in I$ and the two *terminal* vertices, e_{i,V_s} and e_{i,V_t} . For weighted graphs, every edge $e \in E$ has an associated weight w_e . A *cut* $C \subset E$ is a partition of the vertices V of the graph G into two disjoint sets S, T where $V_s \in S$ and $V_t \in T$. The cost of each cut C is the sum of the weighted edges $e \in C$. The minimum cut problem can then be defined as finding the cut with the minimum cost which can be achieved in near polynomial-time [3].

The binary case can easily be extended to a case of multiple terminal vertices. We create two terminal vertices for foreground O and background B pixels for each orientation θ for which $0 \leq \theta \leq \pi$. In our experiments, we have found that choosing the number of orientation labels in the range $N_\theta = [2, 8]$ generates visually acceptable results. Thus the set of labels L has size $|L| = 2 * N_\theta$ and is defined to be $L = \{O_{\theta_1}, B_{\theta_1}, O_{\theta_2}, B_{\theta_2}, \dots, O_{\theta_{N_\theta}}, B_{\theta_{N_\theta}}\}$

4.2 Energy minimization function

Finding the minimum cut of a graph is equivalent to finding an optimal labeling $f : I_p \rightarrow L$ which assigns a label $l \in L$ to each pixel $p \in I$ where f is piecewise smooth and consistent with the original data. Thus, our energy function for the graph-cut minimization is given by

$$E(f) = E_{data}(f) + \lambda * E_{smooth}(f) \quad (4)$$

where λ is the weight of the smoothness term.

Energy data term. The data term provides a per-pixel measure of how appropriate a label $l \in L$ is, for a pixel $p \in I$ in the observed data and is given by,

$$E_{data}(f) = \sum_{p \in I} D_p(f_p) \quad (5)$$

where $D_p(f_p) = \epsilon + \frac{1 - \ln(P(I_p | f(p)))}{1 + (\theta_p - \theta_{f(p)})^2}$.

Energy smoothness term. The smoothness term provides a measure of the difference between two neighbouring pixels $p_i, p_j \in I$ with labels $l_i, l_j \in L$ respectively. Let I_p and I_q be the intensity values in the observed data of the pixels $p, q \in I$ respectively. Similarly, let θ_p and θ_q be the initial orientations for the two pixels. We define a measure

of the *observed* smoothness between pixels p and q as

$$\Delta_{p,q} = \epsilon + \frac{1 + (I_p - I_q)^2}{1 + (\theta_p - \theta_q)^2} \quad (6)$$

The above function favors towards neighbouring pixels with similar intensities and orientations and penalizes otherwise. In addition, we define a measure of smoothness for the global minimization. Let $I_{f(p)}$ and $I_{f(q)}$ be the intensity values under a labeling f . Similarly, let $\theta_{f(p)}$ and $\theta_{f(q)}$ be the orientations under the same labeling. We define a measure of the smoothness between neighbouring pixels p, q under a labeling f as

$$\tilde{\Delta}_{p,q} = \epsilon + \frac{1 + (I_{f(p)} - I_{f(q)})^2}{1 + (\theta_{f(p)} - \theta_{f(q)})^2} \quad (7)$$

Using the smoothness measure defined for the observed data and the smoothness measure defined for any given labeling we can finally define the energy smoothness term as follows,

$$E_{smooth}(f) = \sum_{\{p,q\} \in N} K_{p,q} * \tilde{\Delta}_{p,q} \quad (8)$$

where N is the set of neighbouring pixels, $K_{p,q} = [e^{-\frac{\Delta_{p,q}^2}{2 * \sigma^2}}]$, ϵ is a small positive constant and σ controls the smoothness uncertainty.

The energy function $E(f)$ penalizes heavily for severed edges between neighbouring pixels with similar intensity and orientation, and vice versa, which results in better defined boundaries as shown in the comparison presented in Figure 2.

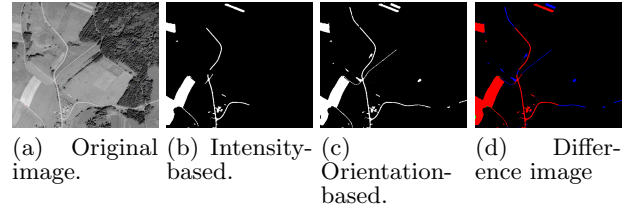


Figure 2: Comparison between intensity- and orientation-based segmentation. Difference image color codes - (red: common points, green: only in inten. segm., blue: only in orient. segm.)

5. ROAD EXTRACTION

5.1 Parallel-line Detection

A bi-modal gaussian-based filter is applied on the curve saliency map returned by the tensor voting to detect parallel-lines. In order to ensure that the parallel-lines correspond to road sides a single mode gaussian-based filter is applied on the segmented road pixels(binary image). This ensures that the area between the parallel-lines is part of a road and not a false positive for example, a road-side and a building's boundary along the side of the road. Three types of information is extracted for each point: the maximum response at each point and, the orientation and width corresponding to the maximum response. This information is then used by the road tracking algorithm to form road segments.

5.2 Road Tracking

Starting from a centerline pixel the road tracking algorithm recursively connects the best neighbouring match. The best match is determined based on the magnitude, orientation and width differences. The result is a set of connected road segments which are finally merged and converted to a vector map.

6. EXPERIMENTAL RESULTS

In our experiments, the smoothness term in 4 was set as $\lambda = 0.25$ and the number of orientation labels used $N_\theta = 8$.

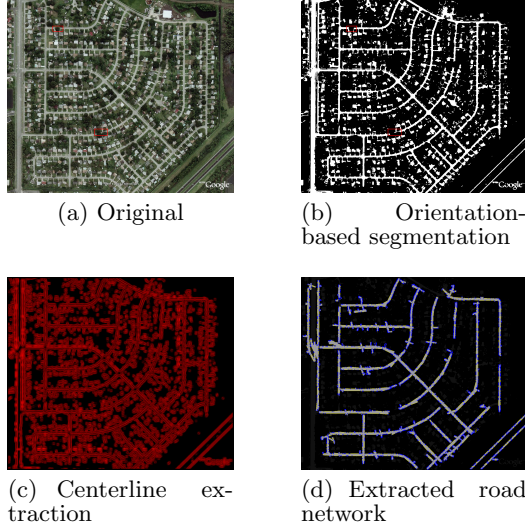


Figure 3: Automatic mode(No user intervention), High resolution satellite image of an urban site with no additional elevation information. In (a),(b) the marked areas show occlusions and shadows.

7. REFERENCES

- [1] A. Baumgartner, C. T. Steger, H. Mayer, W. Eckstein, and H. Ebner. Automatic road extraction in rural areas. In *ISPRS Congress*, pages 107–112, 1999.
- [2] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*, pages 105–112, 2001.
- [3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. In *ICCV*, pages 377–384, 1999.
- [4] J. Canny. A computational approach to edge detection. In *RCV87*, pages 184–203, 1987.
- [5] S. Clode, F. Rottensteiner, and P. Kootsookos. Improving city model determination by using road detection from lidar data. In *IAPRSSIS Vol. XXXVI - 3/W24*, pp. 159–164, Vienna, Austria, 2005.
- [6] R. Deriche. Using canny’s criteria to derive a recursively implemented optimal edge detector. *Inter. J. Computer Vision*, 1(2), May 1987.
- [7] I. R. Fasel, M. S. Bartlett, and J. R. Movellan. A comparison of gabor filter methods for automatic detection of facial landmarks. In *ICAFGR*, pages 231–235, 2002.
- [8] P. Hofmann. Detecting buildings and roads from ikonos data using additional elevation information. In *Dipl.-Geogr.*
- [9] M. Kass. Snakes: Active contour models. *Inter. J. Computer Vision*, 1(4):321–331, 1980. also in *ICCV1*, 1987.
- [10] I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, and A. Baumgartner. Automatic extraction of roads from

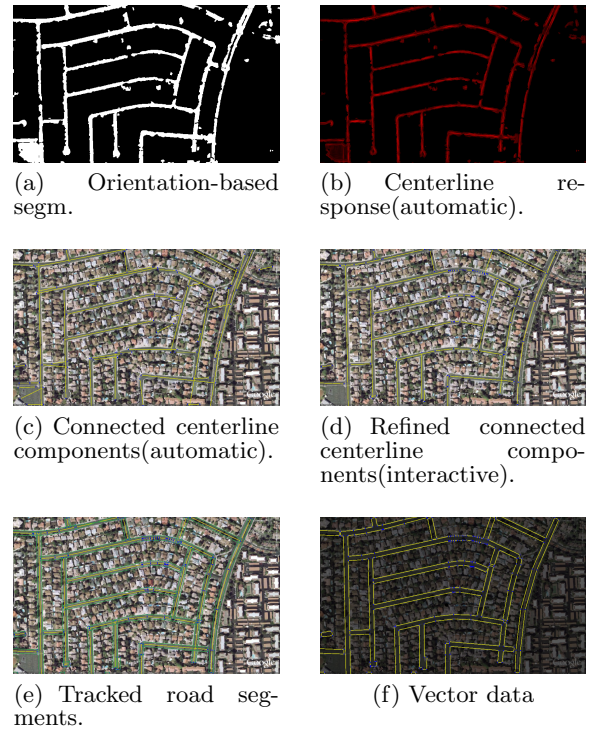


Figure 4: Interactive mode. (a) Segmentation result. (b) Automatic extraction of centerline information. (c) The centerline components recovered automatically. (d) The user refined centerline components. (e) The tracked road segments automatically computed using (d). (f) The final road map vector data.

- aerial images based on scale space and snakes. *Mach. Vis. Appl.*, 12(1):23–31, 2000.
- [11] G. Lisini, C. Tison, D. Cherifi, F. Tupin, and P. Gamba. Improving road network extraction in high-resolution sar images by data fusion. In *CEOS SAR Workshop 2004*, 2004.
- [12] A. Massad, M. Babós, and B. Mertsching. Perceptual grouping in grey-level images by combination of gabor filtering and tensor voting. In *ICPR (2)*, pages 677–680, 2002.
- [13] G. Medioni, M. S. Lee, and C. K. Tang. *A Computational Framework for Segmentation and Grouping*. Elsevier, 2000.
- [14] K. K. Pingle. Visual perception by a computer. In *Automatic Interpretation and Classification of Images*, pages 277–284, 1969.
- [15] L. G. Roberts. Machine perception of 3-D solids. In *Optical and Electro-Optical Info. Proc.*, pages 159–197, 1965.
- [16] C. Steger. An unbiased detector of curvilinear structures. Technical Report FGBV-96-03, Technische Universität München, München, Germany, July 1996.
- [17] B. Wessel. Road network extraction from sar imagery supported by context information. In *ISPRS Proceedings*, 2004.
- [18] C. Zhang, E. Baltsavias, and A. Gruen. Knowledge-based image analysis for 3d road construction. In *Asian Journal of Geoinformatic 1(4)*, 2001.
- [19] Q. Zhang and I. Couloigner. Automated road network extraction from high resolution multi-spectral imagery. In *ASPRS Proceedings*, 2006.