

# URBAN ROAD EXTRACTION VIA GRAPH CUTS BASED PROBABILITY PROPAGATION

Guangliang Cheng, Ying Wang, Yongchao Gong, Feiyun Zhu and Chunhong Pan

National Laboratory of Pattern Recognition,  
Institute of Automation, Chinese Academy of Science  
{guangliang.cheng, ywang, yongchao.gong, fyzhu and chpan}@nlpr.ia.ac.cn

## ABSTRACT

In this paper, we propose a graph cuts (GC) based probability propagation approach to automatically extract road network from complex remote sensing images. First, the support vector machine (SVM) classifier with a sigmoid model is applied to assign each pixel a posterior probability of being labelled as road class, which avoids the weaknesses of hard labels in general SVM. Then a GC based probability propagation algorithm is employed to keep the extracted road results smooth and coherent, which can reduce the connections between roads and road-like objects. Finally, a road-geometrical prior is considered to refine the extraction result, so that the non-road objects in images can be removed. Experimental results on two remote sensing image datasets indicate the validity and effectiveness of our method by comparing with two other approaches.

**Index Terms**— Road extraction, Support Vector Machine, Sigmoid model, Graph cuts, Probability propagation.

## 1. INTRODUCTION

Road extraction from remote sensing images has attracted considerable attention due to its great applications in map update and intelligent transportation systems over the last three decades. Various approaches [1, 2, 3, 4, 5, 6] were proposed to address this issue. In spite of all these attempts, there is no widely recognized solution to the problem. The difficulties mainly lie in two aspects: complicated backgrounds and occlusions due to trees and buildings.

Generally, the existing approaches can be classified into semi-automatic methods and fully automatic methods.

Semi-automatic methods include dynamic programming [1], template matching [2] and active contours [3, 4], etc. Dynamic programming formulates the extraction problem as the minimization of a cost function defined on a graph. Template matching uses an adaptive least square matching algorithm. It can extract road central lines in any orientation. However, these two algorithms mentioned above are a little sensitive to



**Fig. 1.** Comparing results with different methods in the presence of cars and roofs. From left to right: (a) original image, (b) result of method 1 [7], (c) result of method 2 [8], (d) result of our method. Our method is more effective to suppress the spurs and reduce the influence of occlusions due to cars.

seed positions and need a great amount of human interactions. Active contours provide a convenient way of incorporating geometric properties of the roads to guide the extraction process, which is an intuitive approach. Nevertheless, active contours method may not obtain satisfactory results under complex backgrounds and its computing time is somewhat longer than other methods.

Fully automatic methods [5, 6] have drawn considerable attention due to their convenience. While to our knowledge, most of them are only adapted to a particular range of specific conditions. For example, [5] and [6] can not effectively work in the case of occlusions and complex backgrounds. Road characteristics may show significant variations in different situations. Thus it is very intractable to develop a general framework for diverse cases.

To solve the problems presented above, a new automatic road extraction method via graph cuts (GC) based probability propagation is proposed. The main contributions of our approach are highlighted as follows.

1. A soft probabilistic output is employed in support vector machine (SVM) classifier. It can distinguish the degree of its belief that one pixel belongs to road class, which improves the accuracy of the label propagation during the process of the GC based algorithm.
2. A GC based probability propagation algorithm is used to smooth the road areas, which can remove the spurs and make road areas coherent. An example is illustrated in Fig. 1, our results are more coherent and show less spurs.

This research work is supported by National Natural Science Foundation of China under grants 61375024, 91338202, 61305049 and 61331018.

The remainder of this paper is arranged as follows. In Section 2, the proposed methodology is introduced. Section 3 exhibits experimental results and comparisons on two data sets. Conclusions and discussions will be drawn in Section 4.

## 2. THE PROPOSED METHODOLOGY

In this section, we will describe the details about how our approach works. As shown in Fig. 2, given an original image, we aim to extract road network accurately. We first present SVM probabilistic output (see Section 2.1), which plays a great role in extraction process. Then GC based probability propagation algorithm is used to keep the extraction results smooth and coherent (see Section 2.2). Finally a road-geometrical prior is adopted to eliminate those road-like objects (see Section 2.3).

### 2.1. SVM Probabilistic Output

SVM is widely used in binary classification due to its relative high accuracy in the case of high dimension and small training set [9]. These properties are exactly what we need in road extraction. Besides the spectral distinctiveness between road and non-road areas, we also find that road areas are smoother than backgrounds. Thus, image gradients are integrated as discriminative feature for road extraction. We define  $\mathbf{z}_i = (\mathbf{s}_i^T, \nabla_x I_i, \nabla_y I_i)^T$  as the feature vector of pixel  $i$  in image  $I$ , where  $\mathbf{s}_i$  denotes the RGB spectral vector,  $\nabla_x I_i$  and  $\nabla_y I_i$  are the image gradient information in horizontal and vertical coordinate respectively. It should be noticed that all data in  $\mathbf{z}_i$  are normalized to the range of  $[0, 1]$ . A general SVM forces each test sample to have hard labels exactly equal to 1 or -1, which is unsuitable for the classification results of less confidence. To increase the distinction among classification results, a sigmoid model [10] is constructed to map the SVM output to posterior probabilities as follows:

$$P(y_i = 1 | \mathbf{f}_i) = \frac{1}{1 + \exp(\mathbf{w}^T \mathbf{f}_i)}, \quad (1)$$

where  $\mathbf{w} = [\alpha, \beta]^T$ ,  $\mathbf{f}_i = [f_i, 1]^T$ ,  $f_i$  is the general SVM output and  $y_i \in \{-1, 1\}$  is the label.  $\alpha$  and  $\beta$  are unknown parameters of sigmoid function. Given a training set  $\{(f_i, y_i)\}$ , a new training set  $\{(f_i, t_i)\}$  is constructed as,

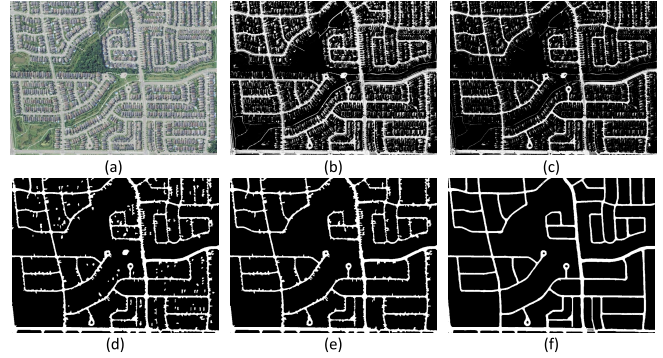
$$t_i = \frac{y_i + 1}{2}, \quad (2)$$

where  $t_i$  is the targeting probability representation.

The parameters  $\alpha$  and  $\beta$  are estimated by minimizing the negative log likelihood of the training data, which is a cross-entropy error function as follows:

$$\min - \sum_i t_i \log(p_i) + (1 - t_i) \log(1 - p_i), \quad (3)$$

where  $p_i$  is defined in Eq. (1) for data  $i$ .



**Fig. 2.** The illustration of processing stages of our method. (a) Original image. (b) Probabilistic map without gradient information. (c) Probabilistic map with gradient information. (d) GC based processing result. (e) Final result. (f) Ground truth.

For the purpose of robustness and avoidance of overfitting, a cross-validation strategy is employed to estimate the parameters of SVM. We implement the probability estimations by applying LibSVM package [11] for convenience.

To verify the effectiveness of the gradient information, a comparison is illustrated in Fig. 2 (c) and (d). As can be seen, (c) is much cleaner and more accurate than (b). Thus, it is necessary to use the gradient information as features.

### 2.2. Graph Cuts Based Probability Propagation

After the previous processing step, we have obtained the pixel-wise probability of road class. Nevertheless, the result of road region lacks of smoothness and coherence. Hence a GC [12, 13] based framework is employed to tackle this problem. This stage can be described as a pixel labeling problem, which is formulated as a cost function:

$$C(\mathcal{L}) = \lambda \cdot C_r(\mathcal{L}) + C_b(\mathcal{L}), \quad (4)$$

where  $\mathcal{L}$  is a labeling set,  $C_r(\mathcal{L})$  and  $C_b(\mathcal{L})$  are regional term and boundary term respectively,  $\lambda$  is a trade-off parameter. The regional term  $C_r(\mathcal{L})$  defines the individual penalty for classifying each pixel into road or non-road class. The boundary term describes the coherence between spatially neighboring pixels.

In graph cuts, an undirected graph  $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$  is constructed, where  $v_p \in \mathcal{V}$  denotes a pixel  $p \in I$  and  $e_{p,q} \in \mathcal{E}$  represents an undirected edge between two neighbouring pixels  $p, q \in I$ . Specifically, the regional term can be defined as follows:

$$C_r(\mathcal{L}) = \sum_{p \in \mathcal{V}} V_{p, L_p}, \quad (5)$$

where  $\mathcal{V}$  is pixel set in the image,  $L_p$  is the label of pixel  $p$ ,  $V_{p, L_p}$  is the cost of assigning pixel  $p$  to label  $L_p$ . It can be computed as:

$$V_{p,L_p} = -\log(Pr_p), \quad (6)$$

here  $Pr_p$  is the probability of assigning pixel  $p$  to the road class, which is obtained from the classifier (as Eq. (1)).

The boundary term can be denoted as:

$$C_b(\mathcal{L}) = \sum_{p,q \in \mathcal{N}} E_{p,q} \cdot m(L_p, L_q), \quad (7)$$

where  $\mathcal{N}$  contains all unordered pairs of neighbouring pixels under a standard 8-neighborhood system.  $E_{p,q}$  measures the difference between two neighbouring pixels as:

$$E_{p,q} = \frac{1}{\|z_p - z_q\|_2 + \epsilon}, \quad (8)$$

where  $z_p$  and  $z_q$  are the feature vectors defined as the previous section.  $\epsilon$  is used to avoid a zero divisor. In the following experiments, we set  $\epsilon = 0.001$ .

In Eq. (7),  $m(L_p, L_q)$  can be denoted as metric distance to measure the cost of labeling smoothness as:

$$m(L_p, L_q) = \begin{cases} 0, & L_p = L_q, \\ 1, & L_p \neq L_q. \end{cases} \quad (9)$$

After the GC based algorithm, most roads can be extracted, as Fig. 2 (d) shows. However, some road-like objects are also improperly detected as roads. Thus a road-geometrical prior is incorporated to eliminate those road-like residues.

### 2.3. Elimination with road-geometrical prior

In urban areas, there are many substances appearing as roads, such as roofs and cars, etc. Compared with roofs, roads are more connected in topology. Thus we define the objects containing above  $M$  pixels as roads. All others are suspected to be non-road class. Empirically, we set  $M = 1500$ . In remote sensing images, roads are elongated ribbons with long length and short width. Thus, this property is used in this work. We use a bounding box to approximate the isolated objects. The length-width ratio is achieved as follows:

$$\text{ratio}_i = \frac{L_i}{W_i}, \quad (10)$$

where  $L_i$  and  $W_i$  denote the length and width of the bounding box for object  $i$  respectively. Any detected objects with bigger ratio than a given threshold is reclassified as the road class. We find it sufficient to fix the threshold as 5.

## 3. EXPERIMENTS

To verify the performance of our method, plentiful experiments as well as comparisons are provided in this section.

### 3.1. Dataset Description

Our datasets were collected from the area of Toronto, Ontario, Canada ( $43^\circ N$ ,  $79^\circ W$ ). Two datasets, each consisting of 5 images, are from different areas to verify our method. All images have a size of at least  $600 \times 600$  pixels with a spatial

resolution of 1.2 m per pixel. All these images have complex backgrounds and occlusions of trees and cars. We randomly sample 3 patches of  $200 \times 200$  pixels from the first image of each dataset as training set. All the images of each dataset constitute the test set. We manually labelled the ground truth of each image as a reference data to evaluate the quality of the methods.

### 3.2. Compared Algorithms

To evaluate the performance, the proposed method is compared with two related methods. The details are listed as follows:

1. Our method: SVM-GC method.
2. Method 1: K-means clustering and morphological operations method [7].
3. Method 2: support vector machine and morphological operations method [8].

It should be noted that codes of the other two methods are not readily available, thus we implement them according to the details described in their papers. Our codes can achieve the same performance as the original paper shows. In the following experiments, we adjust the parameters to gain the best performance of the two approaches.

### 3.3. Parameter Selection

In this paper, SVM with radial basis function (RBF) kernel is separately trained for each dataset. Here two parameters of RBF,  $C$  and  $\gamma$ , are selected by five-fold cross validation. The parameter  $\lambda$  in GC based probability propagation algorithm is tuned empirically. While we find that satisfactory results for two datasets can be obtained when setting  $\lambda = 1$ . Thus we keep the parameter fixed in the following experiments.

### 3.4. Quality Evaluation

Generally, the evaluation of road extraction algorithm can be measured by *completeness*, *correctness* and *quality* defined in [14]. *Completeness* measures the proportion of matched road areas in reference road data. *Correctness* represents the percentage of matched road areas in extracted road data. *Quality* is a combination of *correctness* and *completeness*. They are defined as follows:

$$\text{Completeness} = \frac{TP}{TP + FN}, \quad (11)$$

$$\text{Correctness} = \frac{TP}{TP + FP}, \quad (12)$$

$$\text{Quality} = \frac{TP}{TP + FN + FP}, \quad (13)$$

where TP, FP and FN are the true positive, false positive and false negative, respectively.

### 3.5. Performance Evaluation

To evaluate the performance of our method, we compare our method with two other approaches on two datasets. The evaluation includes two parts: visual comparisons and quantitative comparisons.



**Fig. 3.** Visual comparisons with different methods. There are two rows and five columns of subfigures. Each row shows results on one data set. From left to right: (a) original image, (b) result of method 1, (c) result of method 2, (d) result of our method, (e) ground truth. In (b), (c) and (d), true positive (TP) is marked in green, false positive (FP) in red, false negative (FN) in blue. (Best viewed in color)

Datasets	completeness ( % )			correctness ( % )			quality ( % )		
	Method 1	Method 2	Ours	Method 1	Method 2	Ours	Method 1	Method 2	Ours
Image 1	77.79	76.16	<b>88.38</b>	73.11	76.18	<b>77.23</b>	60.49	61.51	<b>70.11</b>
Image 2	79.20	69.90	<b>79.83</b>	82.21	<b>90.36</b>	90.34	67.61	65.04	<b>73.55</b>
Image 3	81.25	78.43	<b>84.73</b>	71.61	75.96	<b>83.01</b>	61.45	62.84	<b>72.21</b>
Image 4	73.77	70.29	<b>84.61</b>	78.99	<b>87.53</b>	86.15	61.67	63.89	<b>74.48</b>
Image 5	72.99	75.64	<b>84.47</b>	86.82	81.73	<b>87.61</b>	65.71	64.70	<b>75.46</b>
Avg.	77	70.08	<b>84.41</b>	78.54	82.35	<b>84.87</b>	63.39	63.60	<b>73.16</b>
Image 6	76.69	81.03	<b>88.89</b>	74.71	<b>92.01</b>	91.71	60.89	75.71	<b>82.27</b>
Image 7	80.43	66.89	<b>80.97</b>	77.03	90.91	<b>94.88</b>	64.87	62.69	<b>77.57</b>
Image 8	76.69	80.36	<b>88.21</b>	77.53	91.73	<b>92.32</b>	62.75	74.92	<b>82.18</b>
Image 9	<b>86.47</b>	77.65	85.78	70.05	84.77	<b>85.38</b>	65.83	68.15	<b>76.25</b>
Image 10	<b>80.98</b>	67.95	80.50	77.90	86.54	<b>90.74</b>	65.92	62.97	<b>74.68</b>
Avg.	80.25	74.78	<b>84.87</b>	75.44	89.19	<b>91.01</b>	64.05	68.89	<b>78.59</b>

**Table 1.** Quantitative comparisons with two methods in two datasets, where the red values marked in bold are the best. (Best viewed in color)

**Visual comparisons:** In order to display an intuitive comparison, we illustrate the results comparing with ground truth. Fig. 3 shows the comparing results. More FP appears in method 1 and method 2, namely the red parts in Fig. 3 (b1), (c1), (b2), (c2). In addition, our experimental results show less isolated FN (see Fig. 3 (e1),(e2)). Thus the performance of our proposed method is more similar to the ground truth.

**Quantitative comparisons:** Table 1 shows the experimental results on two urban data sets. As we know, *completeness* and *correctness* can be one-sided measurement. We can enlarge the *completeness* at the cost of reducing the *correctness*, and visa versa. The *quality* term combines with the other two criteria, which is a overall desirability. The results of our proposed approach achieve better or comparable performance in terms of two other methods. Especially, our method is consistently better than other methods in *quality* term. In summa-

ry, average *quality* of our method is 10% higher than others, which demonstrates the validity of our approach.

#### 4. CONCLUSIONS AND DISCUSSIONS

In this paper, we have proposed an automatic road extraction method for remote sensing images, which is based on SVM and GC algorithm. Experiments on two urban road data sets validated that our proposed approach achieves better performance both in qualitative and quantitative comparisons.

Our method still has some limitations that we intend to investigate in future. First, in case of severe occlusions, our method can not effectively work. A strategy that is robust to occlusions should be added to our approach. Besides, our approach does not combine road information at different resolutions, which will be incorporated into our following works.

## 5. REFERENCES

- [1] A. Gruen and H. Li, "Semi-automatic linear feature extraction by dynamic programming and LSB-snakes," *Photogrammetric Engineering and Remote Sensing*, vol. 63(8), pp. 985–995, 1997.
- [2] S.R. Park and T. Kim, "Semi-automatic road extraction algorithm from ikonos images using template matching," *Asian Conference on Remote Sensing*, 2001, pp. 1209–1213.
- [3] I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, and A. Baumgartner, "Automatic extraction of roads from aerial images based on scale space and snakes," *Machine Vision and Applications*, vol. 12(1), pp. 23–31, 2000.
- [4] T. Peng, I. H. Jermyn, V. Prinet, and J. Zerubia, "Extended phase field higher-order active contour models for networks," *International Journal of Computer Vision*, vol. 88, pp. 111–128, May 2010.
- [5] P.N Anil and Dr. S. Natarajan, "Automatic road extraction from high resolution imagery based on statistical region merging and skeletonization," *International Journal of Engineering Science and Technology*, vol. 2(3), pp. 165–171, 2010.
- [6] O. Tuncer, "Fully automatic road network extraction from satellite images," in *International Conference on Recent Advances in Space Technologies*, 2007, vol. 14-16, pp. 708–714.
- [7] R. Maurya, S. Shalini, P.R. Gupta, and K.S. Manis, "Road extraction using k-means clustering and morphological operations," *International Journal of Advanced Engineering Sciences and Technologies*, vol. 5(2), pp. 290–295, 2011.
- [8] S. Zhou, J. Gong, G. Xiong, H. Chen, and K. Iagnemma, "Road detection using support vector machine based on online learning and evaluation," in *IEEE Intelligent Vehicles Symposium*, 2010, pp. 256–261.
- [9] Robert Clarke, Habtom W. Resson, Antai Wang, Jianhua Xuan, Minetta C. Liu, Edmund A. Gehan, and Yue Wang, "The properties of high-dimensional data spaces: implications for exploring gene and protein expression data," *Nature Reviews Cancer*, vol. 8, pp. 37–49, January 2008.
- [10] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Adv. Large Margin Classifiers*, 1999, pp. 61–74.
- [11] C.C. Chang and C.J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2(3), pp. 27, 2011.
- [12] Yuri Boykov, Olga Veksler, and Ramin Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, pp. 1222–1239, 2001.
- [13] Yuri Boykov and Marie-Pierre Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images," in *ICCV*, 2001, pp. 105–112.
- [14] C. Heipke, H. Mayer, C. Wiedemann, and O. Jamet, "Evaluation of automatic road extraction," in *International Archives of Photogrammetry and Remote Sensing*, 1997, pp. 47–56.