

# An Approach to Automatic Road Vectorization of Raster Maps

Yao-Yi Chiang and Craig A. Knoblock

*University of Southern California*

*Department of Computer Science and Information Sciences Institute*

*4676 Admiralty Way, Marina del Rey, CA 90292, USA*

*E-mail: [yaoyichi, knoblock]@isi.edu*

## 1 Introduction

Raster maps are widely available for many areas around the globe. The road network that commonly exists in raster maps is an important source of road vector data for the areas for which road vectors are not readily available. To extract the road vector data from raster maps, the first step is to extract the pixels that represent roads from the raster maps (i.e., the road raster layers in the raster maps) and then extract the road vector data from the road pixels. Since the roads usually overlap with other map features and raster maps very often contain noise introduced during the processes of image compression and scanning, the extraction of road pixels from raster maps is difficult. Moreover, for extracting the road vectors from road pixels, previous work commonly uses the thinning operator [6] or line grouping and parallel-line matching techniques [1] to obtain the skeletons of the connected objects composed of road pixels. The thinning operator is robust and requires no parameter tunings; however, the thinned lines (i.e., the skeletons) are usually distorted around the line intersections and hence the extracted road vectors are not accurate, especially when the thinning operator is applied on thick lines. On the other hand, the line grouping and parallel-line matching techniques require manual settings on various parameters to identify the skeletons, such as the maximum difference between the slopes of two line segments to be merged.

In this paper, we present an approach for extracting accurate road vector data from raster maps with no parameter tunings. We first utilize our previous techniques to extract the road pixels from raster maps [4] and generate a road raster layer that contains one-pixel width road lines representing the skeletons of the connected objects composed of the extracted road pixels [5]. Next, we employ the road-intersection template extraction technique [3] to identify road intersections and extract accurate road-intersection templates from the road raster layer. Finally, we trace the pixels of the road raster layer for extracting the road vectors and use the accurate road-intersection templates to avoid the distortions caused by the thinning operator around each intersection.

## 2 Road Pixel and Road Raster Layer Extraction

To extract the road pixels from raster maps, we utilize our technique in [4], which requires very little user input and is capable of handling various types of raster maps, especially scanned maps. Since raster maps often contain thousands of colors, our technique [4] first employs the Mean-shift and K-means segmentation methods to reduce the number of colors in the input raster map and then identifies the road colors from road areas labeled by the users. With the identified road colors, we can generate a color filter to extract the road pixels. Figure 1(a) shows a sample zoning map where the roads are not explicitly drawn but are shown as white areas between the parcels, and Figure 1(b) shows the extracted road pixels (black).

After we extract the road pixels, we identify the road format (i.e., single-line or parallel double-line roads) and the road width using the parallel-pattern tracing technique and then use the morphological operations (e.g., the erosion and dilation operators) to remove solid areas and reconnect lines [2, 5]. The number of iterations for applying the morphological operations is selected using the automatically identified format and road width. Figure 1(c) shows the results after we apply the morphological operations. To generate the road raster layer, we first erode the thick lines shown in Figure 1(c) using a binary erosion operator, which minimizes the distortions caused by the thinning operator that will be applied next. Then, the thinning operator is used to generate the road raster layer that contains one-pixel width road lines. Figure 1(d) shows the extracted road raster layer.

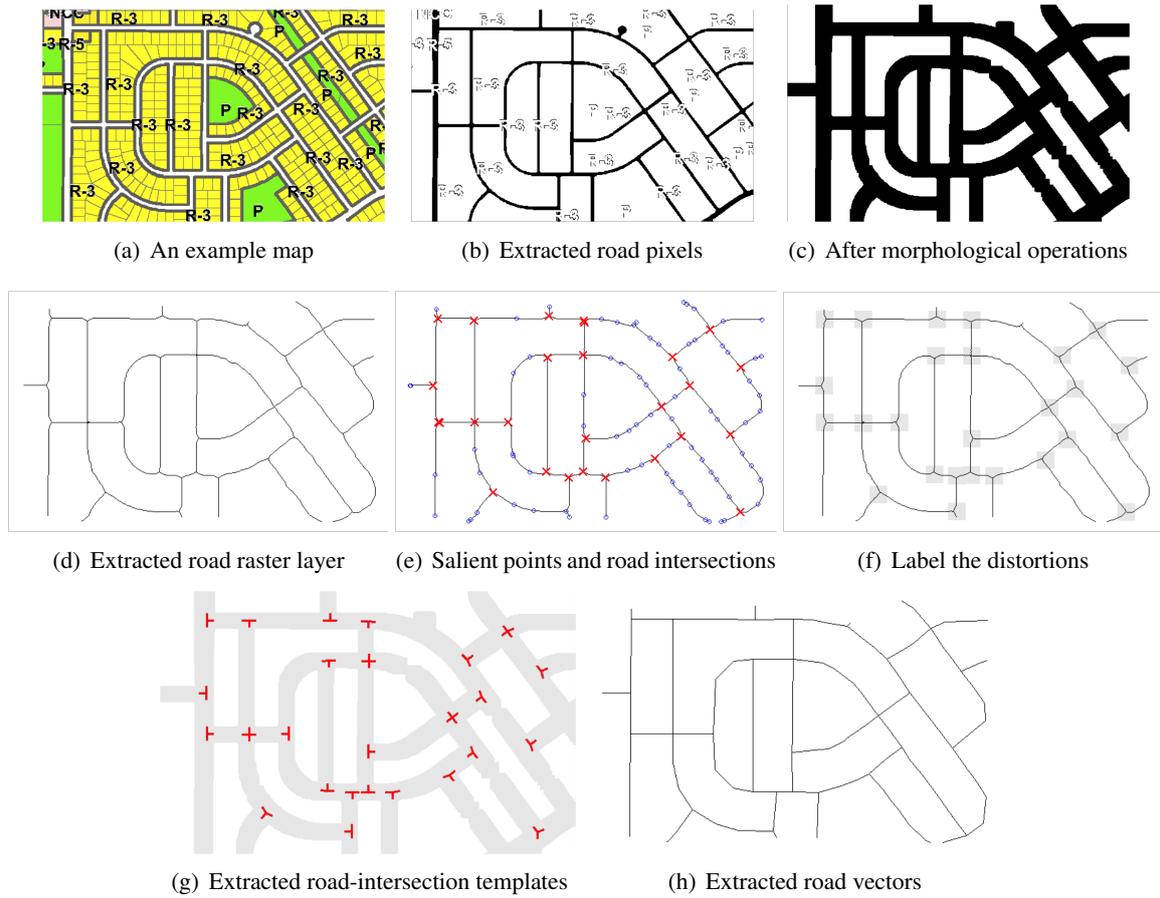


Figure 1: Extract road vector data from an example map

### 3 Road Vectorization

To vectorize the road raster layer, we detect salient points on the one-pixel width road lines using the corner detector [7]. Since there are distortions around the road intersections, the extracted road vectors will not be accurate if we simply use the salient points as the connecting nodes of the road vectors. Therefore, we utilize our technique in [5] to detect road intersections among the salient points. For each salient point, we draw a rectangle around it and then use the number of line pixels that intersects with this rectangle as the connectivity to identify true road intersections. A road intersection must have at least three line segments meet at the intersection (i.e., the connectivity is larger or equal to three). Figure 1(e) shows the detected salient points as circles and the road intersections as X marks. To avoid the distortions and extract accurate road vectors, we utilize our technique in [3] to generate a blob image for labeling the locations of the distorted lines for each identified road intersection as the gray boxes (i.e., the blobs) shown in Figure 1(f). The size of each blob is decided by the detected road format and road width using the parallel-pattern tracing technique [5]. Then, we trace the lines that are not overlapping with the gray boxes to compute accurate orientations for each road that connects to a road intersection. The road orientations are used to update the positions of the road intersections. Figure 1(g) shows the extracted road-intersection templates (i.e., the road intersection positions, road connectivities, and road orientations), and the positions of the road-intersection templates now accurately locate on the medial lines of the thick lines. Finally, we trace the line pixels to connect the salient points and generate the topology of the road network. For the salient points that are identified as road intersections, we use the road-intersection templates to update the position of the salient point so that the road vectors around the road intersections are accurate. Figure 1(h) shows the extracted road vectors. The road vectors around the road intersections are accurate despite the distortions of the one-pixel width road lines shown in Figure 1(f).

## 4 Related Work

Itonaga et al. [6] present a method to extract road vectors from digitally generated maps (i.e., not scanned maps). They employ a stochastic relaxation approach to first extract the road areas and then apply the thinning operator to extract one-pixel width road network. The distortions around the road intersections are corrected based on the straightness of the roads, which is determined using user specified constraints, such as the road width and the differences between the slopes of road lines. In comparison, our approach can be used on a variety of raster maps including scanned maps, and we avoid the distortion using the automatically identified road-intersection positions, which requires no parameter settings.

Bin et al. [1] work on scanned map to extract the road vectors. Instead of using the thinning operator, in [1], the medial lines of parallel road lines are first produced and then linked to generate the road vectors. In general, the vectorization results of utilizing the medial lines can be very accurate for the vectors around the intersections, but the extraction processes require more manual specified parameters than the thinning operator, such as the thresholds to group medial-line segments and to produce the road intersections.

## 5 Discussion

In this paper, we employ the thinning operator to generate the skeletons of road pixels for extracting road vector data from raster maps. We utilize the automatically extracted road-intersection templates to avoid the distortions around the intersections caused by the thinning operator. We show that our technique extracts accurate road vectors from an example raster map with no parameter tunings.

## 6 Acknowledgement

This research is based upon work supported in part by the University of Southern California under the Viterbi School Doctoral Fellowship, and in part by the United States Air Force under contract number FA9550-08-C-0010.

The U.S. Government is authorized to reproduce and distribute reports for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any of the above organizations or any person connected with them.

## References

- [1] D. Bin and W. K. Cheong. A system for automatic extraction of road network from maps. In *Proceedings of the 1998 IEEE International Joint Symposia on Intelligence and Systems*, pages 359–366, 1998.
- [2] R. Cao and C. L. Tan. Text/graphics separation in maps. In *Proceedings of the Fourth International Workshop on Graphics Recognition Algorithms and Applications*, pages 167–177. Springer-Verlag, 2002.
- [3] Y.-Y. Chiang and C. A. Knoblock. Automatic extraction of road intersection position, connectivity, and orientations from raster maps. In *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 1–10, 2008.
- [4] Y.-Y. Chiang and C. A. Knoblock. A method for automatically extracting road layers from raster maps. In *Proceedings of the Tenth International Conference on Document Analysis and Recognition*, 2009.
- [5] Y.-Y. Chiang, C. A. Knoblock, C. Shahabi, and C.-C. Chen. Automatic and accurate extraction of road intersections from raster maps. *GeoInformatica*, 13(2):121–157, 2008.
- [6] W. Itonaga, I. Matsuda, N. Yoneyama, and S. Ito. Automatic extraction of road networks from map images. *Electronics and Communications in Japan (Part II: Electronics)*, 86(4):62–72, 2003.
- [7] J. Shi and C. Tomasi. Good features to track. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1994.