FOURIER SHAPE DESCRIPTORS OF PIXEL FOOTPRINTS FOR ROAD EXTRACTION FROM SATELLITE IMAGES

Jiuxiang Hu, Anshuman Razdan[*]*, John Femiani*

I3DEA, Div of Computing Studies
Arizona State University @Polytech
Mesa, AZ 85212

Peter Wonka, Ming Cui

PRISM, Dept Comp Sci & Eng
Arizona State University
Tempe, AZ 85287

ABSTRACT

In this paper, an automatic road tracking method is presented for detecting roads from satellite images. This method is based on shape classification of a local homogeneous region around a pixel. The local homogeneous region is enclosed by a polygon, called the pixel footprint. We introduce a spoke wheel operator to obtain the pixel footprint and propose a Fourier-based approach to classify footprints for automatic seeding and growing of the road tracker. We experimentally demonstrate that our proposed road tracker can extract the centerlines of roads with sharp turns and intersections effectively, and has relatively small amount of leakage.

Index Terms— Road tracking, pixel footprint, Fourier shape descriptor, satellite image, road extraction.

1. INTRODUCTION

A road network is a graphical representation of the road-structure in a satellite image. Vertices in the graph correspond to sections of roads that may be categorized as linear-segments or intersections according to the number of adjacent road-segments. Road network detection [1][2] can be used for several applications, such as automated correction and updating, registration with multitemporal images for change detection, automatically aligning two spatial datasets, etc. Road tracking is a popular road extraction technique.

Road trackers generally need a starting pixel on the road and another pixel to define the direction of the road, and grow iteratively by searching for and adding line segments. A spanning tree of the road network is formed by some process that uses local information to add new segments to the graph based on the pixel intensities of the image. Mckeown and Denlinger [3] presented a road tracking method based on road cross-section intensity profile matching to follow the direction of road. Vosselman and Knecht [4] improved the road tracking technique by using the Kalman filter. Geman and Jedynak [5] made use of decision tree to control their road tracking to extract major roads in a satellite image. Baumgartner et al. [6] manually enhanced road tracking based on profile matching with a graphical user interface that guides an operator through the whole data acquisition process. However, these techniques often fail to go around the sections of the road narrowed by cars or shadows, and lose their directions at road intersections or segments with high curvature. Manual seeding also makes it difficult to extract road networks fully automatically.

In this paper, we propose a Fourier-based road tracking approach based on pixel footprint classification to automatically extract road networks and detect intersections from satellite images. We introduce a *spoke wheel* operator to obtain a footprint of a pixel in an image (see Section 2), then propose a Fourier shape descriptor to find the directions and size of 'toes' of the pixel footprint for road growing and also to iden-

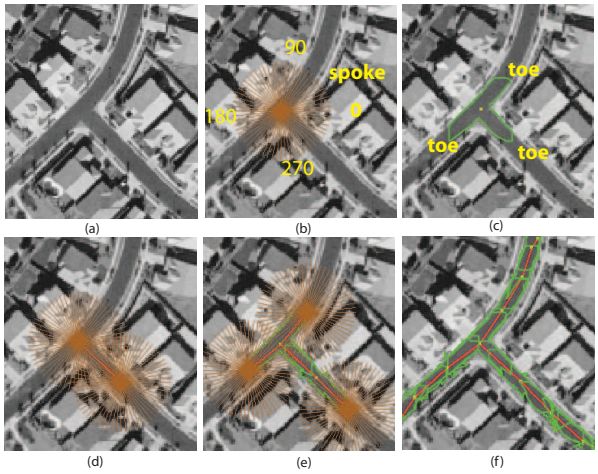

Fig. 1. Main steps of the proposed road tracker. (a)An original image. (b) A spoke wheel with 64 spokes (orange). (c) The footprint (green polygon) around the pixel (yellow dot) with three approximate directional rectangular toes. (d) The proposed road tracker is initialized by $\mathbf{T}_0 = (\mathbf{V}_0, \mathbf{E}_0)$, where $\mathbf{V}_0$ has two vertices and $\mathbf{E}_0$ has one edge (e_0). (e) It is at $\mathbf{T}_1$. (f) The extracted road centerlines $\mathbf{E}_5$ (red line segments) with their pixel footprints (green overlapped polygons).

[*]This paper is in part supported by the National Geospatial-Intelligence Agency under Grant No. HM1582-05-1-2004.

tify the seeds for initializing the proposed road tracker (see Section 3).

2. PIXEL FOOTPRINT AND ROAD TRACKING

2.1. Pixel Footprint

In this section, we introduce the *spoke* and the *spoke wheel* (W) (see Fig. 1(b)) used for finding a *footprint* of a pixel $\mathbf{p}$. A spoke is a line segment of m pixel length. A spoke wheel is a sequence of spokes $\{S_i(\varphi_i, m)\}_{i=0}^{4n-1}$ with common initial pixel $\mathbf{p}$ and evenly spaced angles $\varphi_i = \pi i/2n$. The set of pixels in W centered at the pixel $\mathbf{p}$ with $4n$ spokes is denoted by $W(\mathbf{p}, n, m)$.

The intersection between a spoke and road edge provides useful information to determine the local homogeneous region around a pixel. To search the intersection on a spoke, we start from $\mathbf{p}$, move in the direction of the spoke and observe the absolute intensity differences between $\mathbf{p}$ and pixels along the spoke. We call the first pixel on the ith spoke S_i such that

$$|I(C_i) - I(\mathbf{p})| \geq \sigma(W(\mathbf{p}, n, m)), \quad C_i \in S_i, \qquad (1)$$

the *cutting point* (CP) on S_i, denoted by C_i, where $\sigma(W(\mathbf{p}, n, m))$ is the intensity standard deviation on $W(\mathbf{p}, n, m)$. Notice that $\sigma(W(\mathbf{p_1}, n, m)) \neq \sigma(W(\mathbf{p_2}, n, m))$ usually holds, if $\mathbf{p}_1 \neq \mathbf{p}_2$.

The road associated with each pixel is an anisotropic structure, i.e. the distances, denoted by $\delta(i) = ||C_i - \mathbf{p}||$, in some directions are much longer than those in other directions (see Fig. 1(c)). In order to find the road directions, we connect the CPs on all spokes around a pixel $\mathbf{p}$ in a counterclockwise direction, which results in a closed polygon, called the *footprint* of the pixel $\mathbf{p}$ and denoted by $F(\mathbf{p})$. A footprint usually has several approximate directional rectangular '*toes*' which follow road directions around a pixel (see Fig. 1(c)).

Fig. 2(b) shows 20 footprints of the 20 pixels labeled in Fig. 2(a). The W used has 64 spokes and each spoke is 18 pixels long. The first thirteen footprints have clear directional rectangular toes which correspond to the directions of roads at pixels.

2.2. Road Tracking

Our road extraction method is an iterative line segment growing process based on road footprints, called road tracking (see Fig. 1(d)(e)(f)). Let $\mathbf{P}_i$ be the set of pixels identified as road candidates after iteration i. Let $\mathbf{T}_i = (\mathbf{V}_i, \mathbf{E}_i)$ be a tree representing the road network at iteration i with edges $\mathbf{E}_i$ and vertices $\mathbf{V}_i$. Every vertex $\mathbf{v}_j \in \mathbf{V}_i$ has two types: alive or dead. If $\mathbf{v}_j$ is alive, we determine its footprint, and then search for all possible toes of the footprint by toe-finding algorithm. Because $\mathbf{v}_j$ has exactly one edge in $\mathbf{E}_i$ connecting it to its parent in $\mathbf{V}_i$, it must have at least one toe in the direction of the edge. Let t ($t \geq 1$) be the number of toes of

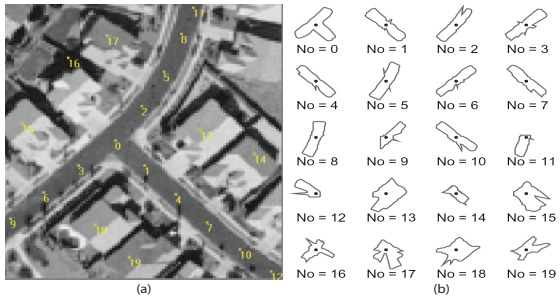

Fig. 2. Illustration of pixel footprints. (a) A zoomed original image with 20 labeled pixels, which all are at the top-left corners of the label box. The first 13 pixels are on road (labeled 0-12). (b) The 20 pixel footprints corresponding to the labeled pixels are obtained. The footprints are drawn at the same scale and in their original orientations. A small circle in every footprint corresponds to the pixel itself. The first 13 footprints corresponding to the pixels on road have clearer rectangular toes than that on roofs or on fields.

$F(\mathbf{v}_j)$. Each toe of the footprint determines a direction and size for road tracking, and generates a new vertex $\hat{\mathbf{v}}_k$, and a new edge $\hat{e}_k = (\mathbf{v}_j, \hat{\mathbf{v}}_k)$ ($1 \leq k \leq t-1$). We always update $\mathbf{P}_{i+1} = \mathbf{P}_i \cup \{\text{pixels enclosed by F}(\mathbf{v}_j)\}$. When $t > 1$, we update $\mathbf{V}_{i+1} = \mathbf{V}_i \cup \{\hat{\mathbf{v}}_k\}$, $\mathbf{E}_{i+1} = \mathbf{E}_i \cup \{\hat{e}_k\}$, and the road tree $\mathbf{T}_i$ is expanded to $\mathbf{T}_{i+1} = (\mathbf{V}_{i+1}, \mathbf{E}_{i+1})$. For each new vertex $\hat{\mathbf{v}}_k$ ($k = 1, \cdots, t-1$), if $\hat{\mathbf{v}}_k \notin \mathbf{P}_i$ then $\hat{\mathbf{v}}_k$ is alive, otherwise $\hat{\mathbf{v}}_k$ is dead, and its position is moved to the centroid of its footprint. Regardless of t, we set the state of the vertex $\mathbf{v}_j$ to dead, and we will not attempt to grow from it again. The road tracking does not stop until all the vertices in a road tree are dead.

There remain two main problems for our road tracking:

- How to start the road tracking process, i.e. how to obtain an edge and two vertices in the initial road tree $\mathbf{T}_0 = (\mathbf{V}_0, \mathbf{E}_0)$.
- How to find toes of a footprint, i.e. how to find the directions and step sizes for the road tracking.

In the following sections, we make use of Fourier shape descriptor of pixel footprint to solve the two problems stated above.

3. FOURIER DESCRIPTORS OF PIXEL FOOTPRINT

3.1. Pixel Footprint Representation

According to the definition of the footprint through a W, it is always true that every ray started from $\mathbf{p}$ intersects the footprint only once. Therefore, we can define a distance between the intersected point and $\mathbf{p}$ as a function of the angle θ of the ray from x-axis, denoted by $\delta_{F(\mathbf{p})}(\theta)$. The distance function $\delta_{F(\mathbf{p})}(\theta)$ is a periodic function with period 2π.

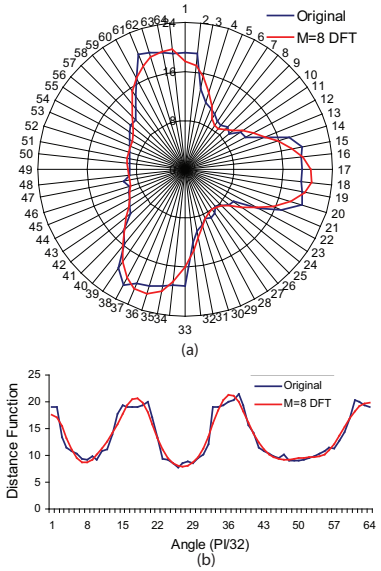

Fig. 3. (a) A 'T'-shaped footprint with three toes (blue curves), (b) its Inverse DFT on low frequency coefficients $M = 16$ (red curves) with three dominating peaks corresponding to toes. There are two small peaks on the top of the dominating peak between 12 and 24 in the blue curve. After low pass filtering, the two peaks merge, and there is only one peak in red curve which corresponds to one of the road directions at the pixel.

3.2. Fourier Shape Descriptor

A pixel footprint obtained above is a closed polygon. It can be represented by a discrete periodic function, $\mathbf{z} = \langle z_0, \cdots, z_{N-1} \rangle$, where $z_l = \delta_F(2\pi l/N)$ is the distance between $\mathbf{p}$ and the lth sampled point ($l = 0, \cdots, N-1$), and N is a integral power of 2. The $\mathbf{z}$ function is then mapped to the frequency domain by way of the discrete Fourier transform (DFT) [7][8]:

$$Z_k = \sum_{l=0}^{N-1} z_l e^{-j\frac{2\pi lk}{N}} = R_k e^{j\Theta_k}, \qquad (2)$$

where $k = -N/2, \cdots, N/2-1$, R_k and Θ_k are the amplitude and the phase of the k^{th} DFT coefficient Z_k, respectively.

Fourier Shape Invariance: Orientation of a road may be any direction. To achieve rotation invariance of footprints, we have to modify the DFT coefficients. Let $\bar{Z}_k = \bar{R}_k e^{j\bar{\Theta}_k}$ be the modified coefficient. In [7], it has been proved that 2D shapes are invariant under rotation, if their DFT coefficients are modified as

$$\bar{\Theta}_k = \Theta_k - \frac{\Theta_{-1} + \Theta_1}{2}. \qquad (3)$$

Dimensional Reduction: Because we only care about the number of toes of a footprint, we use low frequency DFT coefficients which usually retain over 80 percent of the total energy [7]. In particular, we keep only the M ($M << N$) coefficients whose frequency is close to 0. In this paper, we select $M = 16$ from our experiments. It is an *ideal low pass filter*. Although deletion of high frequency coefficients may result in oscillations at bottoms of valleys of the distance function, it keeps the dominating peaks (see Fig. 3).

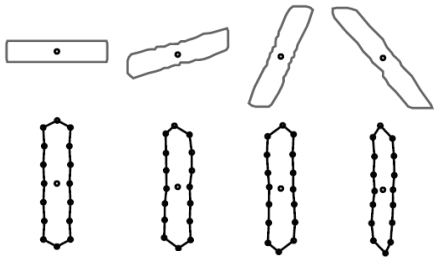

Fig. 4. Top row shows four rectangular footprints with different orientations. Bottom row shows rotation invariance after perform DFT coefficient modification according to Eq. (3).

Fourier Inverse: Using the Inverse DFT, we obtain a normalized footprint, denoted by $\overline{F}$. The discrete $\overline{F}$ is represented by $\bar{\mathbf{z}} =< \bar{z}_0, \cdots, \bar{z}_M >$, which not only satisfies rotation invariances but also performs a low pass filter:

$$\bar{z}_l = \frac{1}{M} \sum_{m=-M/2}^{M/2-1} \bar{Z}_m e^{\frac{2\pi lm}{M}}, \quad 0 \le l \le M \le N. \qquad (4)$$

Proposition 1. *Assume a F approximates a rectangle. Let θ be the angle of its longer edge, generally $\theta \in [0, \pi]$. After modifying the DFT coefficients by using Eq.(3), then the angle of the longer edge of $\overline{F}$ is $\pi/2$,* (see Fig. 4).

3.3. Automatic Seeding

To start road tracking, a seed $\mathbf{T}_0$ is needed. Most road trackers [3][5][6] require manual seeding. Our proposed method retains this option but can also detect seeds automatically. We assume the road network has a segment of low curvature, so at least one footprint is approximately rectangular. In addition, if a F approximates a rectangle, so does its $\overline{F}$ and vice vesa.

To decide if a normalized footprint is rectangular, we would ideally threshold the ratio of its area to that of its minimal oriented bounding box (OBB). By Proposition 1, we use the axis-aligned bounding box ($AABB$) of the normalized footprint as an under-estimate in order to reduce the cost while maintaining invariance to rotation. A footprint F is nearly rectangular, if

$$\frac{\text{AREA}(F)}{\text{AREA}(AABB(\overline{F}))} > 85\%$$

holds. The threshold of 85% was selected by experiment. Once the footprint is determined to be rectangular, we can initialize the proposed road tracker.

Fig. 5. Road network extraction result. (a) An original satellite image. (b) Extracted centerlines of road networks. Different colors were generated from different seeds.

3.4. Toe-finding Algorithm

The toe directions of the footprint correspond to the dominating peaks of its distance function. However, there are small peaks on the top of the dominating peaks of some footprints (see Fig. 3(b)). This makes it difficult to locate dominating peaks. We find the toe directions using $\overline{FT}$, which is more robust because the low pass filter merges these small peaks with their dominating peak. We first find the local maxima that exceed the average of $\bar{\mathbf{z}}$, then merge two successive peaks if the valley between them is not clear[1]. The remaining local maxima are labeled as toes.

4. EXPERIMENTAL RESULTS

The proposed algorithms were tested on high resolution satellite images over urban areas. Fig. 5(a) shows a residential area which has road networks with different widths, shadows, sharp turns and intersections. Automatic seeding was used to extract the road network. Most roads in Fig. 5 are detected.

We compare the Fourier-based toe-finding algorithm with the same method without low pass filter. Fig. 6(a) shows the results using the proposed Fourier-based toe-finding approach. Much less leakage occurs with respect to the results based on the same approach without the low pass filter.

5. CONCLUSION

In this paper, we have presented a new road tracking method based on Fourier shape description of a pixel footprint for extracting road network in satellite images. The low frequency DFT coefficients of the pixel footprint have been used to find the direction and size of its toes. We have detailed how rotation invariant properties can be obtained by properly normalizing DFT coefficients to automatically initialize the proposed road tracking and have experimentally demonstrated that the proposed road tracker can extract roads with sharp turns and intersections effectively and has little leakage. It is also fully automatic versus road trackers based on road cross-section intensity profile matching.

[1]This is examined by calculating the average distance for the horizontal axis value between the two peaks.

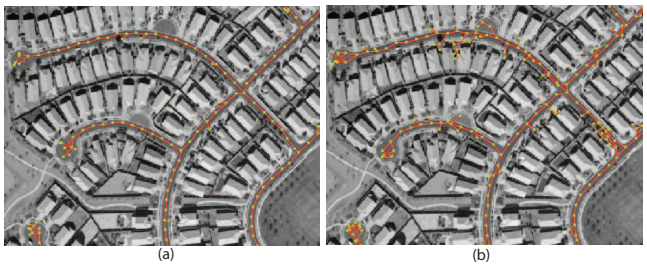

Fig. 6. Road extraction results (a) with Fourier-based toe-finding approach, (b) with non Fourier-based toe finding approach.

6. REFERENCES

[1] X. Jin and C. H. Davis, "An integrated system for automatic road mapping from high-resolution multi-spectral satellite imagery by information fusion," *Information Fusion*, vol. 6, pp. 257–273, 2005.

[2] M. Negri, P. Gamba, G. Lisini, and F. Tupin, "Junction-aware extraction and regularization of urban road networks in high-resolution sar images," *IEEE Trans. Geoscience and Remote Sensing*, vol. 44, no. 10, pp. 2962 – 2971, 2006.

[3] D.M. McKeown and J.L. Denlinger, "Cooperative methods for road tracking in aerial imagery," *Proceedings CVPR '88*, pp. 662–672, 1988.

[4] G. Vosselman and J. Knecht, "Road tracing by profile matching and kalman filtering," *In: Automatic Extraction of Man-Made Objects from Aerial and Space Images, Birkhuser Verlag, Basel, Boston, Berlin*, pp. 265–274, 1995.

[5] D. Geman and B. Jedynak, "An active testing model for tracking roads in satellite images," *IEEE Trans. PAMI*, vol. 18, no. 1, pp. 1–14, 1996.

[6] A. Baumgartner, S. Hinz, and C. Wiedemann, "Efficient methods and interfaces for road tracking," *IEEE Proc. IGARSS'03*, vol. 2, pp. 1059– 1061, 2003.

[7] I. Bartolini, P. Ciaccia, and M. Patella, "Warp: Accurate retrieval of shapes using phase of fourier descriptors and time warping distance," *IEEE Trans. PAMI*, vol. 27, pp. 142–147, 2005.

[8] H. Kauppinen, T. Seppanen, and M. Pietikainen, "An experimental comparison of autoregressive and fourier-based descriptors in 2d shape classification," *IEEE PAMI*, vol. 17, no. 2, pp. 201–207, 1995.