In: Wagner W., Székely, B. (eds.): ISPRS TC VII Symposium – 100 Years ISPRS, Vienna, Austria, July 5–7, 2010, IAPRS, Vol. XXXVIII, Part 7A

Contents   Author Index   Keyword Index

URBAN ROAD TRACKING BY FUSION OF SVDD AND REGION ADJACENCY GRAPHS FROM VHR IMAGERY

Z. Liu[a, *], X. Lin[a,b], J. Zhang[a], P. Pu[a]

[a] Key Laboratory of Mapping from Space of State Bureau of Surveying and Mapping, Chinese Academy of Surveying and Mapping, No.28, Lianhuachixi Road, Haidian District, Beijing 100830, China - (email:zjliu @casm.ac.cn)
[b] School of Resources and Environment Science, Wuhan University, No.129, Luoyu Road, Wuchang District, Wuhan 430079, China

KEY WORDS: Vector, Data, Tracking, Fusion, Imagery

ABSTRACT:

Road surfaces are seriously disturbed by a variety of noises on the very high resolution (VHR) remotely sensed imagery in urban areas, e.g., abrupt geometric deformation and radiometric changes caused by sharp turning, shadows of tall buildings, and appearance of vehicles, which leads to frequent failures for most of current road tracking methods. In this paper, a semi-automatic method is proposed for urban road tracking on VHR imagery. Initially, a human operator inputs three seed points on a selected road, and then necessary information, such as road direction, road width, start point, and a reference template, is automatically derived. The automatic tracking is consequently triggered. During the process, the reference template is moved to generate several target templates. For each target template, a binary template is derived by classifying the target template using support vector data description (SVDD). Subsequently, region adjacency graphs (RAG) is used to eliminate the small disturbing features on the road surfaces in each binary template, which is helpful to search the optimal road centerline points. The above tracking process is repeated until a whole road is completed. Two VHR images were used for the test. The preliminary results show that our method can extract roads more robustly than existing least-squares template matching method in urban areas.

1. INTRODUCTION

The increasing availability of commercial very high resolution (VHR) satellite imaging sensors such as QuickBird, GeoEye-1 and TerraSAR, demands the availability of suitable automatic interpretation tools to extract and identify cartographic features (Lin et al., 2009). Roads are one of the most important cartographic features, and automatic extraction of them is meaningful for various applications such as Geographic Information System (GIS) database updating, transportation analysis and urban planning (Huang and Zhang, 2009). Nevertheless, attempts on developing fully automatic road extraction method for VHR digital imagery have been made for decades (e.g. Hinz and Baumgartner, 2003; Song and Civco, 2004; Jin and Davis, 2005). It still involves several major scientific and technical challenges (Mena, 2003). Therefore, despite a lot of research work on fully automatic approaches for road extraction, the desired high level of automation could not be achieved by now and even in the near future (Baumgartner et al., 2002). One more practical solution to this problem is to adopt a semi-automatic approach that retains the "human in the loop" where the computer vision algorithms are used to assist human extracting the roads (Zhou et al., 2006). Currently, dozens of semi-automatic methods are proposed for road extraction from VHR imagery, and many of them reach various levels of success. In general, these semi-automatic approaches may be grouped into two categories: path optimizers and road trackers or path finders (Amo et al., 2006). A path optimizer is applied to determine an optimal trajectory between manually selected seed points, and it is often realized by improving the dynamic programming and snakes or active contour model (Gruen and Li, 1997) for VHR images. In these models, geometric and radiometric characteristics of roads are integrated by a cost function or an 'energy' function, and then the road extraction is equivalent to seeking the global energy minimum. Amo et al. (2006) improved the active contour model by the region competition algorithm to extract the ribbon roads on aerial images. Dal Poz and do Vale (2003) made a modification of merit function of the original dynamic programming approach, which is carried out by a constraint function embedding road edge properties. However, it is hard to define the reasonable 'energy' function for each road on each VHR image.

Compared to the path optimizers, path finders are more popular. A path finder is an iterative line growing process: it often starts with some seed points, then the local information is used to add new segments into the road network based on the pixel intensities of the image, and typically a human operator is needed to help the path finder go through the various types of image noises such as cars and shadows. For example, McKeown and Denlinger (1988) described one of the most general road finder based on the cooperation between the intensity profile correlation of road cross sections and road edges following. Vosselman and Knecht (1995) imposed the profile matching by using least squares template matching and Kalman filter. Baumgartner et al. (2002) also presented a human-computer interactive prototype system by the above method. Similarly, Zhou et al. (2006) used two profiles, one perpendicular to road direction and the other parallel to road direction, to enhance robustness of the tracker and applied extended Kalman filter and particle filter to solve profile matching issues for road tracking. Slightly different from the above methods; Kim et al. (2004) employed a rectangular reference template of road surfaces to track roads by least squares template matching, and road path is modelled as similarity transform; Hu et al. (2004) presented a road finder using a piecewise parabolic model and least-squares template matching; Zhou et al. (2007) utilized one on-line learning

---

[*] Corresponding author

method based on the principle of one-class support vector machines (SVMs) to find the optimal matched template in road tracking; and Lin et al. (2009) described a road finder by both tracking the lane markings and road surfaces based on least-squares template matching. However, most of the above road finders fail when they encounter the road intersections. Another road finder presented by Hu et al. (2007) can well extract the intersections besides the general roads, and it employed a spoke wheel operator to obtain the road footprints. Despite most of the above road finders perform well on some kinds of roads or intersections such as highways or rural roads where the road surfaces are relatively homogenous on VHR images, they often failed to extract the roads where the surfaces suffer from abrupt geometric deformation and radiometric changes caused by sharp turning, shadows of tall buildings, and appearance of vehicles etc.

How to decrease the negative effects of various types of image noises is a key step to increase the robustness of a path finder. Actually, most of existing path finders, such as McKeown and Denlinger's one (1988), make use of least-squares template matching in searching an optimal road centreline point, but this type of method, using the squared sum of the grey value differences as a measure, is easily impacted by the image noises mentioned above. In this case, new features of roads should be selected and utilized. For example, Zhao et al. (2002) utilized the template matching on a classified image in a semi-automatic road tracking system, and Lin et al. (2009) proposed a novel road signature measure called "parallelepiped angular texture signature" to semi-automatically track roads based on the unique characteristic of roads on a classified sub-windows. It is testified that the supervised classification can indeed provide a novel feature for road tracking. However, most of the existing conventional supervised classification analyses may depict multiple classes including buildings, water, trees etc. besides the roads and they assume implicitly that the set of training sample for each class is large enough (Foody et al., 2006). However for road tracking application, our interest is only focused on just one specific class, road, and the training set size is not large enough in road mapping. Recently, statistical learning theory and one-class SVMs have been used in road extraction from VHR images, e.g., Zhou et al.'s method (2007) mentioned above. The support vector data description (SVDD) is a one-class classifier based on the principles of the SVM, and it provides a very simple to use supervised classification analysis that requires only the training data for the class of interest (Sanchez-Hernandez et al., 2007). Moreover, the accuracy of SVDD classification was considerably higher than that derived from a conventional multi-class parametric classification (e.g., Maximum Likelihood) and popular alternatives (e.g. feedforward neural networks) (Sanchez-Hernandez et al., 2007).

In this paper, SVDD is employed to track road on VHR images for the first time. Particularly, once training samples are provided from the reference template derived from road initialization, SVDD is trained and used to identify road pixels in sub-windows generated in moving of reference template, and, subsequently region adjacency graphs (RAG) is employed to eliminated the image noises contained by road surfaces in the classified sub-windows, and template matching is utilized to determine the optimal road direction for road tracking.

2. RELATED COMPUTER ALGORITHMS

One-Class   Classification   by   SVDD
Road is the only specific class of interest in road mapping from VHR imagery, and a range of approaches exist to classify a specify class of interest, including reconstruction methods (e.g. Pizzi et al., 2001), density methods (e.g. Fumera et al., 2000), and boundary methods (e.g. Zhou et al., 2007). However, reconstruction methods and density methods require extensive knowledge and large amount of information about the data set of interest. Fortunately, boundary methods are more feasible in that they do not require the extensive knowledge of the data set, as they concentrate on the boundary that fits around the class of interest (Tax, 2001). This benefit makes the boundary methods very attractive to use in remote sensing applications (Sanchez-Hernandez et al., 2007). Boundary methods are largely based the statistical learning method (Vapnik, 1995) and the principles of SVMs (Song and Civco, 2004), and the recently developed SVDD is comparable to SVMs. The basic idea of SVMs binary classifier that seeks to fit an optimal separating hyperplane or decision boundary between the classes; however, the SVDD searches for a closed boundary around the training data, namely a hypersphere, instead of looking for a hyperplane (Tax and Duin, 2004).

The hypersphere may be defined by $F(R,O) = R^2$, where O is the centre and R is the radius. Therefore, the problem SVDD attempts to solve is to find the hypersphere with the constraints that all the training data x_i are within R^2. Figure 1 shows the geometrical interpretation in a two dimensional case. The problem can be formulated as follows:

$$\text{Minimize } (F(R,O) = R^2)$$

$$\text{subject to } \| x_i - O \| \le R^2, \qquad \forall_i. \qquad (1)$$

In order to allow the possibility of outliers in the training set, the distance from x_i to the center O should not be strictly smaller than R^2 but larger distance should be penalized. In this sense, slack variables $\varepsilon_i \ge 0$ must be introduced into the error function and, correspondingly, the above optimization problem changes into:

$$\text{Minimize } (F(R,O,\varepsilon) = R^2 + C\sum_i \varepsilon_i)$$

$$\text{subject to } \| x_i - O \| \le R^2 + \varepsilon_i, \quad \varepsilon_i \ge 0, \quad \forall_i \quad (2)$$

where C is a known and given coefficient that makes a trade-off between the volume of the description and the misclassification errors.

Using the technique of Lagrange multipliers, this optimization problem can be formulated into the following quadratic programming problem:

$$\underset{\alpha,\gamma}{Max}\,\underset{R,O,\varepsilon}{Min}\, L(R,O,\alpha,\gamma,\varepsilon) = R^2 + C\sum_i \varepsilon_i -$$

$$\sum_i \alpha_i \times \{R^2 + \varepsilon_i - (\| x_i \|^2 - 2x_i \cdot O + \| O \|^2)\}$$

$$- \sum_i \gamma_i \varepsilon_i$$

In: Wagner W., Székely, B. (eds.): ISPRS TC VII Symposium – 100 Years ISPRS, Vienna, Austria, July 5–7, 2010, IAPRS, Vol. XXXVIII, Part 7A

Contents   Author Index   Keyword Index

subject to $\alpha_i \geq 0$, $\gamma_i \geq 0$, $\sum_i \alpha_i = 1$, $O = \sum_i \alpha_i x_i$, C -

$$\alpha_i - \gamma_i = 0, \quad \forall_i. \tag{3}$$

where α_i, γ_i are Lagrange multipliers respectively, and $\underset{\alpha,\gamma}{Max} \underset{R,O,\varepsilon}{Min}$ means $L(R,O,\alpha,\gamma,\varepsilon)$ should be minimized with respect to R, O, ε and maximized with respect to α, γ. Substituting the last three constraints into the target function will give the following simplified formula:

$$\text{Maximize } L = \sum_i \alpha_i (x_i \cdot x_i) - \sum_{i,j} \alpha_i \alpha_j (x_i \cdot x_j)$$

$$\text{subject to } 0 \leq \alpha_i \leq C, \sum_i \alpha_i = 1, \quad \forall_i \tag{4}$$

Note that the magnitude of the Lagrangian multiplier α_i varies with the position of the case relative to the hypersphere. Figure 1 shows a case within the hypersphere where $\alpha_i = 0$; a case on the hypersphere boundary where $0 < \alpha_i < C$; and a case outside the hypersphere where $\alpha_i = C$. Moreover, only the samples with $0 < \alpha_i < C$ are the support vectors of the description (Sanchez-Hernandez et al., 2007), which are essential for the calculation of the optimal hypersphere with centre O and radius R.

The solution of SVDD is given by:

$$O = \sum_i \alpha_i x_i. \tag{5}$$

While the decision function for the SVDD classification is given by:

$$f(x) = \| x - O \|^2 \leq R^2 \tag{6}$$

For the non-linear case, as with SVMs, noticing the training data appeared in the optimization problem in the form of dot products, a mapping ϕ of the data using the kernel functions (Commonly-used kernel functions refer to Song and Civco, 2004) may be firstly denoted as:

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j). \tag{7}$$
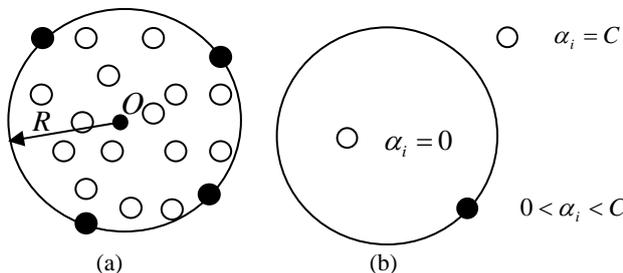
Figure 1. Support vector data description. (a) Hypersphere containing the target data. The shaded objects on the edge of the sphere are the support vectors. (b) Magnitude of the two Lagrange multipliers for cases inside, on the boundary, and outside the hypersphere.

In this paper, a human operator will provide the training data for the SVDD classifier by a reference template, as being introduced in the next section, and the classification will be performed on a patch of the image. Moreover, in the SVDD classification, the polynomial-degree function kernel of free parameter value two and $C = 0.01$ was selected as done by (Sanchez-Hernandez et al., 2007).

Noise Removal by RAG

The above SVDD classification procedure produces a patch of binary image, which labels pixels belonging to road class as 1 while the other non-road pixels as 0. Some parts of road surfaces may be misclassified into non-road class due to the various types of image noises such as occlusions of vehicles, shadows of trees and buildings, as shown in Figure 3. If we suppose that any image primitives, belonging to non-road class, contained in road class polygons are road class primitives, these primitives should be reclassified into road class. Herein, RAG, as shown in Figure 3, is employed to do the topological analysis and reclassify the noises on road surfaces into road class, which will significantly decrease the side-effects of the image noises. Figure 3 illustrates that the road surfaces are dilated and the noises are eroded after the RAG analysis, which decrease the negative effects of noises.

3. SYSTEM FRAMEWORK

In our semi-automatic system, a human operator is required in the road extraction process where computer algorithms are utilized to assist the operator performing measurement tasks. From the user's point of view, the procedure is as follows: the operator first inputs three seed points that detect a short segment of a road which serves as initialization for computer algorithms, and then the proposed algorithms are launched and automatically track the road axis as long as possible. Whenever the internal evaluation of the algorithms indicates that the tracker might have lost the road centreline, the system needs intervention of the user. Then the operator has to confirm the path finder (tracker) or he/she must edit the extracted road and put the tracker back the road again. Concretely, the system is based on the following road model and the main procedures.

Suppose a road model is represented as a queue of road centreline points that is denote as:

$$\{(x_0, y_0, \theta_0), (x_1, y_1, \theta_1), \cdots, (x_i, y_i, \theta_i), \cdots, (x_n, y_n, \theta_n)\}$$

where (x_i, y_i) are the planar coordinates of the ith road centreline point while θ_i is the corresponding direction of the above road point, and the relationship between the ith point and the (i-1)th point can be expressed as:

$$\begin{bmatrix} x_i \\ y_i \\ \theta_i \end{bmatrix} = \begin{bmatrix} x_{i-1} + L_{step} * \cos\theta_{i-1} \\ y_{i-1} + L_{step} * \sin\theta_{i-1} \\ \theta_{i-1} \end{bmatrix} (i \geq 1) \tag{8}$$

where L_{step} is the suitable step size of the increment (i.e., the distance between two consecutive points on the road axis), and L_{step} is set to road width in this paper. As a result, if (x_0, y_0, θ_0) is known, then the automatic road tracking is equivalent to searching the optimal direction for each road point. Particularly, the semi-automatic tracking is divided into the following steps.

Step 1: Initialization by three seed points

Similar to Vosselman and Knecht's method (1995), the initialization is also accomplished by manually selected seed points. However, we take another strategy in which a three consecutive mouse clicks strategy is adopted to obtain the starting point, direction, width of the road, and the step size as well. This three seeds method is feasible for most of the roads on VHR images, and it is accomplished as follows (see Figure 2): the human operator enters a road segment with two

consecutive mouse clicks on A' and B with the axis joining the points defining one road sideline A'B, which indicates road 9direction arctangent(A'B), then the following third click on C, on the other roadside, defines the width w of the road. w is equal to the distance between the point C and the line A'B. As a result, the above three points can also derive a rectangle A'B'B"A" with width w and length l. Particularly, the direction of A'B' is equal to arctangent(A'B) while C is located on the side B"A", and $l = 2 * w$. Then a start point A, denoted as (x_0, y_0), of the road is derived from the middle point of A' A". The above information forms the first road centreline point (x_0, y_0, θ_0) in the road model mentioned above where $\theta_0 = $ arctangent(A'B), and obtain the next road centreline point (x_1, y_1, θ_1) by Equation 8 where $\theta_1 = \theta_0$, add the above two points into the road queue sequentially. Simultaneously, the pixels in the template A'B'B"A" also serves as training samples for the SVDD classifier, and then a predict model is derived by training the SVDD classifier.
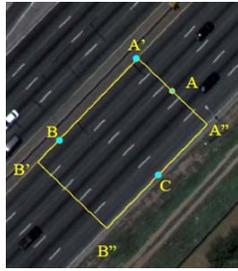
Figure 2. Road initialization by three seed points

Step 2: Acquire the next road axis candidate point

Take the two latest road centreline points out from the current road queue, denoted as $(x_{i-1}, y_{i-1}, \theta_{i-1})$ and (x_i, y_i, θ_i), respectively. Revolve around the pixel p (x_i, y_i), and form a square $S(\ p(x_{TopLeft}, y_{TopLeft})$, $p(x_{BottomRight}, y_{BottomRight})\ )$, where $p(x_{TopLeft}, y_{TopLeft})$ and $p(x_{BottomRight}, y_{BottomRight})$ are the top left corner and bottom right corner of the square respectively. Concretely, the coordinates of two corners are calculated by the following formula:

$$x_{TopLeft} = x_i + \sqrt{2} * l * \cos(\theta_{i-1} - \frac{3}{4}\pi)$$

$$y_{TopLeft} = y_i + \sqrt{2} * l * \sin(\theta_{i-1} - \frac{3}{4}\pi)$$

$$x_{BottomRight} = x_i + \sqrt{2} * l * \cos(\theta_{i-1} + \frac{3}{4}\pi) \qquad (9)$$

$$y_{BottomRight} = y_i + \sqrt{2} * l * \sin(\theta_{i-1} + \frac{3}{4}\pi)$$

where l is the length of the reference rectangle introduced in the first step.

As mentioned above, the reference A'B'B"A" derives a predict model for the SVDD classifier, and then perform SVDD classification on the above obtained squared subset image. Subsequently, set the pixels of road subclass as 1, meanwhile set the pixels of any other subclass as 0, and then perform the RAG analysis on the binary image to reclassify the image noises on the road surfaces into road class, which will decrease the negative effects of various types of noises.

Following, at each road centreline point p (x_i, y_i), a rectangular template with width w and height l is revolved on the classified image, and $T(\alpha, w, h, p)$ is defined as the mean for the rectangular set of pixels of around pixel p whose principal axis lies at an angle of α from the road direction θ_{i-1}. This measure is computed for a set of angles $\alpha_0, \ldots \alpha_n$ at pixel p (x_i, y_i). Angles $\alpha_0, \ldots \alpha_n$ are with same interval δ. At the point p, the mean of the template at each rotating angle forms a set of values $\{T(\alpha_0, w, h, p), T(\alpha_1, w, h, p), \ldots, T(\alpha_n, w, h, p)\}$, named as classified angular texture signature (CATS). Figure 3(a) shows a CATS with $\delta = 5°$. The direction of the significant maximum which has a minimal inclination with road direction θ_{i-1} is taken as the real direction of current road axis point, and replace θ_{i-1} with the optimal value.

Step 3: Validate the above optimal point
Once the above obtained point is added into the road model, check whether any stopping criterion is fulfilled as follows:

- the change of the directions of two adjacent road segments is larger than predefined threshold T;

- the minimal mean value of the optimal template surpass T_1;

- compactness of CATS polygon is larger than T_2;

- approaching an extracted road or border of the image.

To find the relationship between the shape of the CATS polygon and corresponding pixel types, we plot the CATS values around the pixel under consideration with corresponding direction and link the last point to the first point. The resulting polygon is called the CATS polygon, and Figure 3(e) shows the calculated CATS for pixel p with the CATS polygons. If the road has a good contrast with its surrounding objects, the polygon usually looks like an ellipse or ∞-shape, or a circle in other cases. The compactness of CATS can be defined as the compactness of the CATS polygon using Equation (10):

$$CATS_{compactness} = \frac{4\pi \cdot A}{P^2} \qquad (10)$$

where A and P are the area and perimeter of the CATS polygon, respectively. It is employed to check whether the shape of the CATS polygon looks like a circle. A circle-like CATS polygon usually indicates that the tracker is no longer fit for tracking the road ahead. Note that our program will calculate the compactness of CATS at regular intervals to verify whether the CATS is still suitable for tracing a road.

If any of these conditions is encountered, exit the tracking procedure and go to Step 4). Otherwise, obtain the next road centreline point $(x_{i-1}, y_{i-1}, \theta_{i-1})$ by equation (8) and add this point into the road queue, and go to Step 2 again.

Step 4: Stop the automatic following

If no rule can be made to continue the tracking procedure, the system will stop tracking, report the reason, and offer an appropriate choice of user interaction. The user can then modify the traced path with the aid of common GIS-functionalities, manually digitize complex roads, update the reference template (occurrence of change of the number of lanes, or significant change of spectral characteristics due to

In: Wagner W., Székely, B. (eds.): ISPRS TC VII Symposium – 100 Years ISPRS, Vienna, Austria, July 5–7, 2010, IAPRS, Vol. XXXVIII, Part 7A

Contents   Author Index   Keyword Index

different ages, construction materials, illumination angles, etc.), or restart the tracking process from the next specified location.

(a) A rotating rectangular template and its resulted sub-windows

(b) Resulted sub-window   (c) Classified image by SVDD

(d) Image noises removal by RAG (e) A CATS polygon
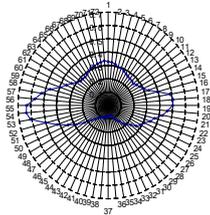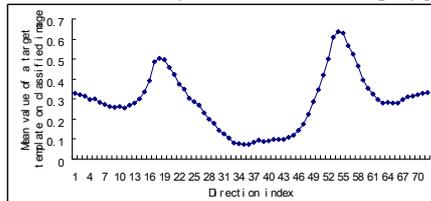
(f) Values of the CATS in image (a)

Figure 3. Road direction determination by SVDD and RAG

4. EXPERIMENTS AND PERFORMANCE EVALUATION

A prototype system, based on our proposed method and rectangular template matching (Kim et al. 2004) is developed in VC++6.0 IDE under Win-XP OS. Note that the implementation SVDD is based on Tax's code in matlab environment (Tax, 2001) and the standard one-class SVM in LibSVM (Chang and Lin, 2001).

Two airborne images in urban areas were tested to verify the capabilities of each road tracker. The roads on the above two images are disturbed by various types of image noises such as zebras, occlusions of vehicles, material change, and the extracted results are shown in Figure 4 and Figure 5, respectively. For the first image, the existing rectangular template matching method is feasible, but it failed at the sharp turning and the intersection, and it also failed to track the ring road around the stadium in the second image due to large change of radiometric characteristic of the road. Fortunately, the proposed method succeeds to extract the accurate centrelines of roads in the above two images. The above two tests suggest that our proposed method is more robust to various types of image noises such as sharp turnings, road intersections, zebras, vehicles and material change etc.

(a) Result of rectangular template matching

(b) Result of our method

Figure 4. Extracted roads at an intersection

5. CONCLUSIONS

This paper presents a semi-automatic system for road tracking from VHR remotely sensed imagery. Once a human operator input three seed points that derive a reference template, the system adopts a new combination strategy to automatically track the road networks. Particularly, in the automatic tracking process, SVDD classifier is employed to produce a patch of classified binary image based on the reference template, RAG is utilized to erode the various types of image noises and enhance the road feature space on the binary image, and template matching using mean of the values of the pixels in a target template is used to search the optimal road direction and next road centreline point. The above procedure is repeated until a whole road is tracked. At the same time, a human operator is retained in the tracking process to supervise the extracted results, to response to the program's prompts. Experiments are performed to extract roads from aerial/satellite imagery. The results show that our proposed road trackers can more robustly extract most of the main roads than other typical road trackers, which have significant practical applications. Future work will also include the optimization of the algorithms to speed up the calculations.

In: Wagner W., Székely, B. (eds.): ISPRS TC VII Symposium – 100 Years ISPRS, Vienna, Austria, July 5–7, 2010, IAPRS, Vol. XXXVIII, Part 7A

Contents   Author Index   Keyword Index

Figure 5. The extracted ring road of a stadium by our proposed method (Note that the rectangular template matching method failed on this image)

REFERENCES

Amo, M., Martínez, F., and Torre, M., 2006. Road extraction from aerial images using a region competition algorithm. *IEEE Transactions on Image Processing*, 15(5), pp. 1192-1201.

Baumgartner, A., Hinz, S., and Wiedemann, C., 2002. Efficient methods and interfaces for road tracking. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 34, Part 3B, pp. 309-312.

Chang, C.C., and Lin, C.J., 2001. LIBSVM: a library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm (accessed 20 July 2007).

Dal Poz, A.P., and do Vale, G.M., 2003. Dynamic programming approach for semi-automated road extraction form medium-and high-resolution images. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXIV, Part 3/W8, pp. 87-91.

Foody, G.M., Mathur, A., Sanchez-Hermandez, C., and Boyd, D.S., 2006. Training set size requirements for the classification of a specific class. *Remote Sensing of Environment*, 104, pp. 1-14.

Fumera, G., Roli, F., and Giacinto, G., 2000. Reject option with multiple thresholds. *Pattern Recognition*, 33(12), pp. 2099–2101.

Gruen, A., and Li, H., 1997. Semi-automatic linear feature extraction by dynamic programming and LSB-Snakes. *Photogrammetic Engineering and Remote Sensing*, 63(8), pp. 985-995.

Hinz S., and Baumgartner, A., 2003. Automatic extraction of urban road network from multi-view aerial imagery. *ISPRS Journal of Photogrammetry & Remote Sensing*, 58, pp. 83-98.

Hu, X., Zhang, Z., and Tao, C.V., 2004, A robust method for semi-automatic extraction of road centrelines using a piecewise parabolic model and least squares template matching. *Photogrammetric Engineering and Remote Sensing*, 70(12), pp. 1393-1398.

Hu, J., Razdan, A., Femiani, J.C., Cui, M., and Wonka, P., 2007. Road network extraction and intersection detection from aerial images by tracking road footprints. *IEEE Transaction on Geoscience and Remote Sensing*, 45(12), pp. 4144-4157.

Huang, X., and Zhang, L.P., 2009. Road centreline extraction from high-resolution imagery based on multi-scale structural features and support vector machines, *International Journal of Remote Sensing*, 30(8), pp. 1977-1987.

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

Kim, T., Park, S.R., Kim, M.G., Jeong, S., and Kim, K.O., 2004. Tracking road centerlines from high resolution remote sensing images by least squares correlation matching. *Photogrammetric Engineering and Remote Sensing*, 70(12), pp. 1417-1422.

Lin, X., Liu, Z., Zhang, J., and Shen, J., 2009. Combining multiple algorithms for road network tracking from multiple source remotely sensed imagery: a practical system and performance evaluation. *Sensors*, 9(2), pp. 1237-1258.

Mena J.B., 2003. State of the art on automatic road extraction for GIS update: a novel classification. *Pattern Recognition Letters*, 24(16), pp. 3037- 3058.

Pizzi, N. J., Vivanco, R.A., and Somorjai, R.L., 2001. EvIdent: A functional magnetic resonance image analysis system. *Artificial Intelligence in Medicine*, 21(1–3), pp. 263–269.

Sanchez-Hernandez, C., Boyd, D.S., and Foody, G.M., 2007. One-class classification for mapping a specific land-cover class: SVDD classification of Fenland. *IEEE Transactions on Geoscience and Remote Sensing*, 45(4), pp. 1061-1073.

Song, M., and Civco, D., 2004. Road extraction using SVM and image segmentation. *Photogrammertric Engineering and Remote Sensing*, 70(12), pp. 1365-1371.

Tax, D.M.J., 2001. One-class classification. Ph.D. dissertation, Delft University Technology, Delft, The Netherlands.
Tax, D.M.J., and Duin, R.P.W., 2004. Support vector data description. *Machine Learning*, 54(1), pp. 45–66.

Vapnik, V., 1995. The nature of statistical learning theory. New York: Springer-Verlag.

Vosselman, G., and Knecht, D.J., 1995. Road tracing by profile matching and Kalman filtering. In: *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, A. Gruen, O. Kuebler and P. Agouris (Eds)(Basel, Switzerland: Birkhauser Verlag), pp. 265–274.

Zhao, H., Kumagai, J., Nakagawa M., and Shibasaki, R., 2002. Semi-automatic road extraction from high resolution satellite images. In: *Proceedings of ISPRS Photogrammetry and Computer Vision*, Graz, Ustrailia, pp. A-406.

Zhou, J., Bischof, W.F., and Caelli, T., 2006. Road tracking in aerial images based on human-computer interaction and Bayesian filtering. *ISPRS Journal of Photogrammetry & Remote Sensing*, 61(2), pp. 108-124.

Zhou, J., Li, C., and Bischof, W.F., 2007. Online learning with novelty detection in human-guided road tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 45(12), pp. 3967-3977.

ACKNOWLEDGEMENTS