Overcoming occlusions in road extraction from high resolution satellite images

Kambam Sreekanth Reddy[1]
Krishnan Sundara Rajan[2]

Lab for Spatial Informatics
International Institute of Information Technology, Hyderabad, India-500032
[1]sreekanth.kambam@research.iiit.ac.in , [2]rajan@iiit.ac.in

Abstract. Road extraction from the high resolution satellite imagery is one of the important problems in the remote sensing area. One of the major challenges in the road extraction is overcoming of the occlusions. However, little research has been conducted in handling the occlusions in the roads. In this paper, an algorithm for handling the occlusions is presented that is based on a previously developed semi automatic adaptive texture matching approach (ATM-R). ATM-R uses the seeds provided by the user for detecting the road areas and extracts only contiguous road areas. Most of the road tracking algorithms stop when encountered with an occlusion and are unable to work around them. So they require many seeds to be given by the user to extract the road if there are many occlusions in the image. The occlusion handling algorithm proposed here can jump over the occlusions and can proceed with road tracking, resulting in using minimum number of user given seeds by being able to handle the partial and fully occluded areas. The algorithm is tested on different high resolution satellite images and its performance on IKONOS image is presented and discussed in this paper.

Keywords: Remote Sensing, Image Processing, Contiguity, ATM-R, Forward Looking, Multiple Lookup

1. Introduction

Road network database is one of the essential features for various GIS applications like traffic management, transportation management and urban planning. Road map can be generated by an on-field survey, GPS and also by using aerial or satellite images. In low resolution satellite images, road may appear as lines and road extraction problem can be thought of extracting elongated line features. But in the high resolution images road width or road area is clearly visible and they no longer appear as lines, instead they appear as elongated narrow areas with parallel borders. In the recent years, many works are being done to extract the roads from the high resolution satellite imagery. Many road extraction methods based on template matching (Park and Kim, 2001 and Lin et al., 2011), segmentation (Song and Civco, 2004 and Grote et al., 2007), road tracking from the initial seeds are developed. Based on the involvement of the user, these techniques can be categorized into automatic, semi-automatic and manual.

In Haitao et al. (2008), a method based on the LSB-snake model is presented. A self-adapting template matching assists the LSB-snake model to provide with enough number of seeds to extract the road. In Anil and Natarajan (2010), an automatic road extraction method based on statistical region merging and skeletonization is presented. Statistical region merging is used in segmenting the image and the road extraction is done using skeleton pruning of the regions obtained by discrete curve evolution. In Rajeswari et al. (2011), a method based on level set and normalized cuts is discussed. The main idea of the method is to remove the areas which are road-like features and then apply level set and normalized cut methods to model the roads as boundaries and extract roads by formulating the problem as boundary evolution problem. In Vinay Pandit (2009), a method based on adaptive texture matching(ATM-R) is presented. A new road region is detected using a set of reference road regions and a set of target regions and

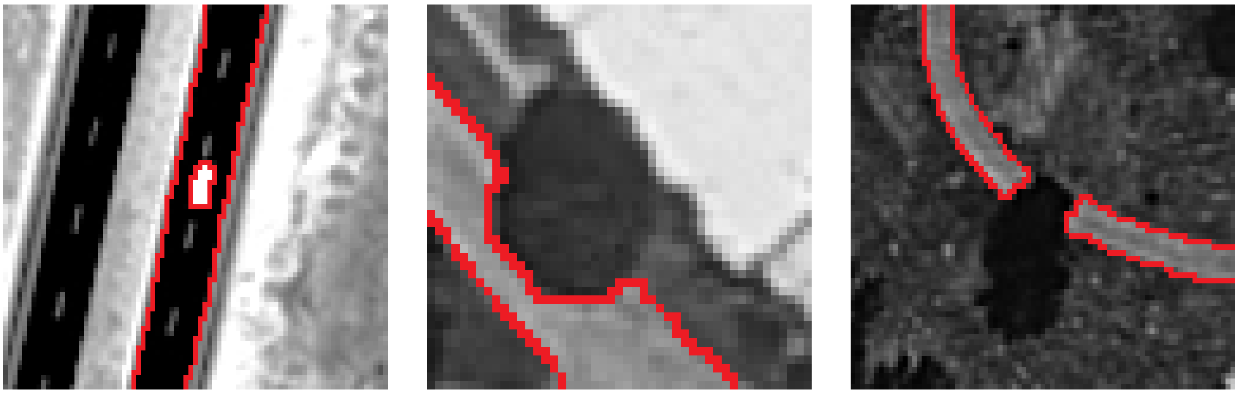

Figure 1: Different kind of Occlusions. (a) Island Occlusion. (b) Partial Occlusion. (c) Full Occlusion.

this process is repeated to detect the further road regions.

The major limitations of these algorithms are the storage requirement and time complexity for the larger images and the need to give a large number of user seeds to extract the road when they are many occlusions present in the image. Occlusion is defined as any region which breaks the contiguity of the road feature being extracted from the image. In the Figure 1(c), the ATM-R needs two seeds on either side of the occlusion to extract the road. Thus more number of occlusions, implies more number of the seeds has to be provided by the user. In roads, the occlusions can occur mainly due to vehicles, trees, shadows of buildings etc. The categorization of these occlusions can be helpful in dealing with them suitably. Occlusions can be categorized in to two types geometrically.

- Regular-shaped occlusions: Mostly these type of occlusions are due to man-made structures like vehicles, shadows of the buildings with geometrically well-defined boundaries.

- Irregular-shaped Occlusions: Mostly these type of occlusions are due to natural elements like trees, their shadows etc.

Occlusions can be categorized into two types spectrally, based on how they differ with the spectral features of the road.

- Spectrally similar: The objects causing the occlusion have spectral values similar to spectral features of the road.

- Spectrally different: The objects causing the occlusion have different spectral values from the spectral values of the road.

For the purposes of road feature extraction, the occlusions are broadly classified into three categories based on how they disturb the road extraction algorithms.

- Island occlusion: If a road is split into two and merges after some distance, it creates an island of non-road region. That region is considered as island occlusion.

- Partial occlusion: If the occlusion formed modifies the geometry of the road without breaking the contiguity of the road, then that occlusion is considered as partial occlusion.

- Full occlusion: If the occlusion breaks the contiguity of the road, then that occlusion is considered as full occlusion.

Island occlusions are generally caused mainly due to vehicles whereas partial and full occlusions are mainly due to objects in the neighbourhood of the roads like trees, buildings and their shadows. All the three kind of occlusions are shown in the Figure 1. The aim of the paper is to overcome the occlusions in the road extraction with the minimum number of seeds while handling the partial and full occlusions of the road features.

2. Methodology

Before discussing the algorithm which handles the occlusions, a brief account of the ATM-R algorithm is presented here, which is then modified to achieve the goal of overcoming occlusions.

2.1. ATM-R algorithm

ATM-R algorithm is a adaptive texture based matching algorithm in which new road regions are detected with the help of already detected regions. The detected regions are utilized as the reference regions for the further detection. Initially the algorithm starts with the seeds provided by the user. The rectangular road region is represented by its centroid g and its orientation angle θ. To detect a new road region, a set of last p adjacent reference regions is considered.

$$R = [r_{(g_1,\theta_1)}, r_{(g_2,\theta_2)}, ..., r_{(g_p,\theta_p)}] \qquad (1)$$

where each region is of length l, width h and $\{g_1, g_2, ..., g_p\}$ and $\{\theta_1, \theta_2, ..., \theta_p\}$ are the centroids and orientations of the p regions respectively. A set of $2 * t + 1$ target rectangular regions are considered to find a new road region.

$$S = [s_{(g_{mid},\theta_p)}, s_{(g_{\alpha_1},\theta_p)}, s_{(g_{\alpha_2},\theta_p)}, ..., s_{(g_{\alpha_t},\theta_p)}, s_{(g_{\beta_1},\theta_p)}, s_{(g_{\beta_2},\theta_p)}, ..., s_{(g_{\beta_t},\theta_p)}] \qquad (2)$$

where $s_{(g_{mid},\theta_p)}$ is the region which is at a distance of length l from the reference region $r_{(g_p,\theta_p)}$ along the length in the orientation θ_p. $\{s_{(g_{\alpha_1},\theta_p)}, s_{(g_{\alpha_2},\theta_p)}, ..., s_{(g_{\alpha_t},\theta_p)}\}$ are the regions which are at a distance of $\{1, 2, ..t\}$ pixels respectively along the width to the left side of $s_{(g_{mid},\theta_p)}$. Similarly $\{s_{(g_{\beta_1},\theta_p)}, s_{(g_{\beta_2},\theta_p)}, ..., s_{(g_{\beta_t},\theta_p)}\}$ are the regions which are at a distance of $\{1, 2, ...t\}$ pixels respectively along the width to the right side of $s_{(g_{mid},\theta_p)}$. The illustration shown in the Figure 2 explains the locations and interactions between the various regions.

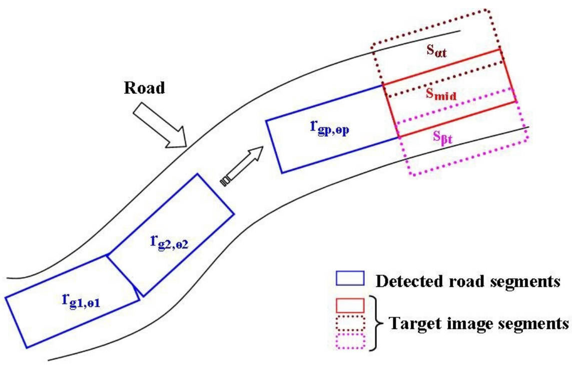

Figure 2: Reference and Target rectangular regions.

The features used for comparing the reference and target regions are the tamura features, k-means based features and histogram features. Before placing the new region in the pool of road regions, orientation of the region is corrected using the method suggested by Haralick and Shapiro (1992). In order to detect the road regions at the road intersections and to track the diverted roads, the tracking is done not only along the length but also in orthogonal directions. For detailed description of the ATM-R algorithm, reader can refer to Vinay Pandit (2009).

2.2. Occlusion handling algorithm

As mentioned earlier, ATM-R or similar algorithms requires huge amount of seeds if they are many occlusions in the image. The ATM-R stops tracking the road whenever the contiguity of the road patches is broken by an obstruction. The occlusion handling algorithm proposed here instead of stopping, looks for the road region beyond the obstruction. If any road patch is detected on the other side of the obstruction, then that obstruction is assumed to be an occlusion. The algorithm also assumes that the length of the occlusion it handles is relatively small compared to the road length. In this paper partial and full occlusions of length less than a predefined threshold λ are handled. The occlusion handling algorithm (ATM-ROH) is explained in Algorithm 1.

Algorithm 1: Occlusion handling algorithm (ATM-ROH)

$R = \{r_1, r_2, ..., r_p\} \leftarrow p$ reference road regions
$\theta_p \leftarrow$ orientation of the p_{th} reference region
$s_{mid} \leftarrow$ region at a distance of l from the p_{th} region along the length in the orientation θ_p
$\{s_{\alpha_1}, s_{\alpha_2}, ..., s_{\alpha_t}\} \leftarrow$ regions at a distance of $\{1, 2, ..., t\}$ pixels along the width to the left side of s_{mid}
$\{s_{\beta_1}, s_{\beta_2}, ..., s_{\beta_t}\} \leftarrow$ regions at a distance of $\{1, 2, ..., t\}$ pixels along the width to the right side of s_{mid}
$S = \{s_{mid}, s_{\alpha_1}, s_{\alpha_2}, ..., s_{\alpha_t}, s_{\beta_1}, s_{\beta_2}, ..., s_{\beta_t}\} \leftarrow 2 * t + 1$ target regions
$flag1 = compare(R, S)$
if $flag1 \leftarrow 1$ **then**
    $p + 1_{th} region \leftarrow$ the road region detected.
else
    $i \leftarrow 1$
    while $i <= 3$ **do**
        $u_{i_{mid}} \leftarrow$ region at a distance of $(i + 1) * l$ from the p_{th} region along the length in the orientation θ_p
        $\{u_{i_{\alpha_1}}, u_{i_{\alpha_2}}, ..., u_{i_{\alpha_t}}\} \leftarrow$ regions at a distance of $\{1, 2, ..., t\}$ pixels along the width to the left side of $u_{i_{mid}}$
        $\{u_{i_{\beta_1}}, u_{i_{\beta_2}}, ..., u_{i_{\beta_t}}\} \leftarrow$ regions at a distance of $\{1, 2, ..., t\}$ pixels along the width to the right side of $u_{i_{mid}}$
        $U_i = \{u_{i_{mid}}, u_{i_{\alpha_1}}, u_{i_{\alpha_2}}, ..., u_{i_{\alpha_t}}, u_{i_{\beta_1}}, u_{i_{\beta_2}}, ..., u_{i_{\beta_t}}\} \leftarrow 2 * t + 1$ target regions
        $flag2 = compare(R, U_i)$
        if $flag2 \leftarrow 1$ **then**
            $p + 1_{th} region \leftarrow$ the road region detected.
            $i \leftarrow 4$
        else
            $i = i + 1$
        end if
    end while
end if

The first set of target regions are the adjacent regions of the last p_{th} reference region. At an obstruction no road region is found in this set. So the algorithm considers a new set of target regions which are the adjacent regions of the previous target set. This kind of forward tracking or multiple lookup of the patches is repeated. If a road region is found in any of the lookup patches, then the old target sets which are present in between the p_{th} reference region and the current target set are considered as an occlusion.

Since the occlusion handling algorithm does multiple lookup of the patches only three times, the occlusions of length less than threshold $\lambda = 3 * l$ are handled. Because of this forward tracking, the algorithm is able to extract the road using minimum number of seeds. The illustration of the regions U_1, U_2, U_3 are shown in the Figure 3, with S and U_1 being the occluded region of the road.

2.3. Post Processing

The proposed algorithm is able to detect the full occlusions, but can fail to detect some kind of partial occlusions. If the road region adjacent to the partial occlusion along the width is not detected as a road patch by the algorithm due to its dimensions then that occlusion is treated in the same way as a full occlusion is treated and that kind of partial occlusions can be detected. A sample of this kind of partial occlusion is shown in Figure 4(a). But if that road region is detected as a road patch, then that partial occlusion is not detected as an occlusion because the contiguity is not broken in the tracking of the road. A sample of this kind of partial occlusion is shown in Figure 4(b). This special kind of occlusions is detected by running a post processing step. The extracted road is processed once again for checking the presence of this kind of occlusions. If the width of the extracted road abruptly decreases for some distance and increases after, then the region is assumed to have this kind of partial occlusion. The average width of the road on either side of the occlusion is used to estimate the width at the partial occlusion area and the estimated width is used in filling it.

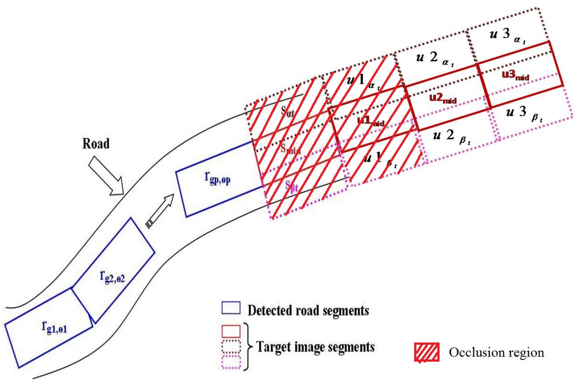

Figure 3: Multiple lookup patches

3. Results and Discussion

3.1. Data

The occlusion handling algorithm has been tested on different satellite imagery. Here the result on an IKONOS panchromatic image of dimensions 575x800 which contains the different kinds of occlusions is shown. The IKONOS panchromatic image is of spatial resolution 0.8m and its radiometric resolution is 11bits. The maximum length of the occlusion in the input image is 19 (in pixels) and minimum length is 5. The input image is shown in the Figure 5(a).

3.2. Results and Discussion

To validate the occlusion handling algorithm the following metrics are used

- Reduction in the number of seeds: Difference between the number of seeds required by ATM-R (NSA) and the number of seeds required by the ATM-ROH (NST)
- Number of true negatives (NTN)

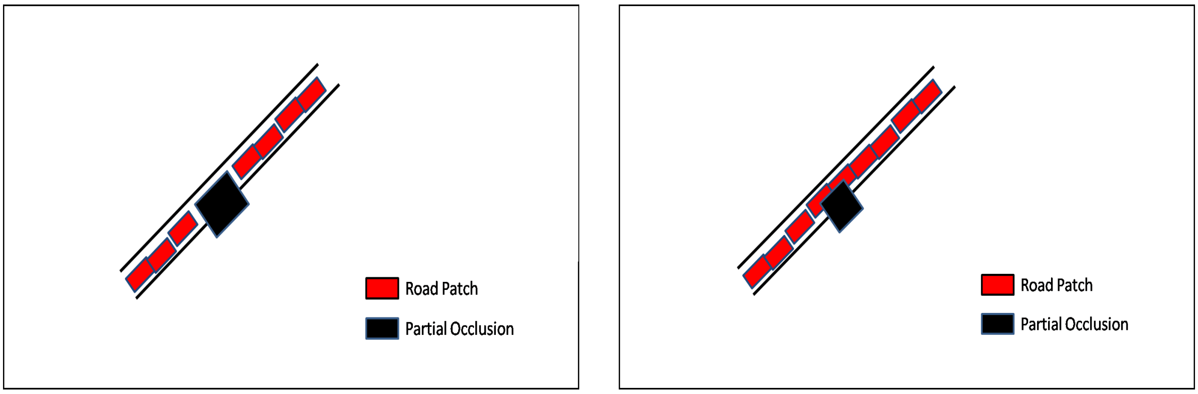

Figure 4: (a) Partial Occlusion handled by ATM-ROH. (b) Partial occlusion not handled by ATM-ROH but handled in the post-processing step.

Table 1: Results

Image	NSA	NST	NTN	NFP	% of road extracted
Figure 5(a)	8	2	2	5	77

- Number of false positives (NFP)

For the input image 5(a), the ATM-ROH algorithm is able to extract the road, using just two user seeds, where as the original ATM-R algorithm is able to extract the same percentage of the road using eight seeds. Table 1 shows the different metrics calculated for the given input image. Figure 5(b) shows the occlusions detected by the ATM-ROH algorithm, marked with green patches. Only two occlusions marked with pink arrows are not detected by the algorithm. The pink arrow marked occlusion on the left side is not detected because the length of the occlusion is greater than the threshold λ and the other pink marked occlusion on the right side is not detected because the occlusion is on the curvature of the road. Rest all occlusions are detected by the ATM-ROH algorithm. This shows that the algorithm is efficient enough to detect all the occlusions which are not on the curvature of the road and of length less than threshold λ. For a clear understanding of the result, the input image and the result of the ATM-ROH algorithm is zoomed onto one of the occlusion and is shown in the Figure 6.

Out of seventeen occlusions detected in the given image, five are the false matches, shown with red and yellow arrows as seen in the Figure 5(b). Three of five false matches, shown with red arrows, are because of the wrong orientation of these patches. Since their orientation is different from that of the road, major part of these patches have non-road pixels and they are not detected as a road patch and the patch next to the false match is detected as a road patch. So these patches are assumed and detected as occlusions by the algorithm. The other two false matches, shown with yellow arrow, is due to the spilling of detection of the road patches by the ATM-R algorithm into the neighbourhood of the road. The spilling is because the neighbourhood patches also have the similar spectral and textural characteristics as that of the road. Because of the spilling, these false matches are detected in the immediate neighbourhood of the road.

4. Conclusions and Future Work

In this paper an algorithm for handling the occlusions in the road extraction is presented with encouraging results. The algorithm is able to detect all the partial and full occlusions which are not on the curvature of the road and of length less than the threshold. The partial occlusions which are not detected are handled in the post processing step. The algorithm also

Figure 5: (a) Input image (b) The detected occlusions in green patches. The true positives are not marked with any arrow. The true negatives are marked with pink arrow. The false postives are marked with red and yellow arrows.

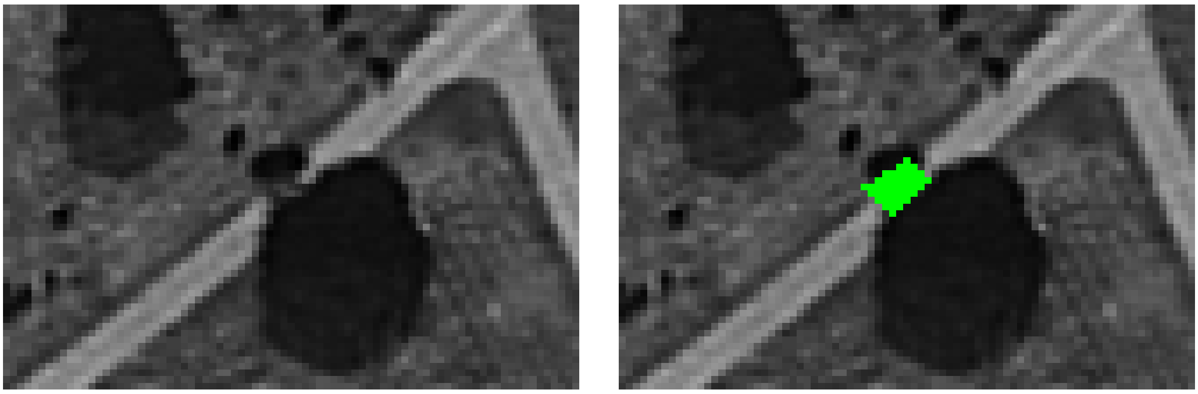

Figure 6: (a) The zoomed part of the input image. (b) Detected occlusion in the green patch.

detects some false patches as occlusions because of the wrong orientation of these patches and due to the spilling of the road into the neighbourhood of the road. The algorithm in its current form is not suitable to detect the island occlusions. The future work is to overcome some of these limitations of the ATM-ROH algorithm and also to improve the ATM-R algorithm in reduction of the spilling of the detection into the neighbourhood and to better estimate the extracted road area.

Acknowledgements

IKONOS image used in this experiment are sourced from the sample CD of CARTERRA product of Space Image Inc.

References

Park, S. R.; Kim, T. Semi-automatic road extraction algorithm from IKONOS images using template matching. In: Proc. ACRS 2001 - 22nd Asian Conference on Remote Sensing, Singapore, 5-9 November 2001, Vol. 2, p. 1209-1213. Available at: <http://www.crisp.nus.edu.sg/~acrs2001/pdf/293PARK.pdf>. Accessed on: 14 November 2012.

Lin, X.; Zhang, J.; Liu, Z.; Shen, J. Semi-automatic extraction of road networks by least squares interlaced template matching in urban areas. **International Journal of Remote Sensing**, v. 32, n. 17, p. 4943-4959, 2011.

Song, M.; Civco, D. Road extraction using SVM and image segmentation. **Photogrammetric engineering and remote sensing**, v. 70, n. 12, p. 1365-1371, 2004.

Grote, A; Butenuth, M.; Heipke, C. Road extraction in suburban areas based on normalized cuts. In: Photogrammetric Image Analysis, Munich, 19-21 September 2007. Available at: <http://www.isprs.org/proceedings/XXXVI/3-W49/PartA/papers/51_pia07.pdf> - International archives of Photogrammetry and remote sensing, Vol. XXXVI, part 3-W49A. Munich. Accessed on: 14 November 2012.

Haitao, Z.; Zhou, X.; Qing, Z. Research on road extraction semi-automatically from high resolution remote sensing images. In: XXI ISPRS Congress, Beijing, 3-11 July 2008. Available at: <http://www.isprs.org/proceedings/XXXVII/congress/3b_pdf/101.pdf> - International archives of Photogrammetry and Remote Sensing, Vol. XXXVII, part B3b. Beijing. Accessed on: 14 November 2012.

Anil, P. N.; Natarajan, S. Automatic road extraction from high resolution imagery based on statistical region merging and skeletonization. **International Journal of Engineering Science and Technology**, v. 2, n. 3, p. 165-171, 2010.

Rajeswari, M.; Gurumurthy, K. S.; Pratap, R. L.; Omkar, S. N. Automatic road extraction based on normalized cuts and level set methods. **International Journal of Computer Applications**, v. 18, n. 7, p. 10-16, 2011.

Vinay, P. Automatic road extraction from high resolution satellite imagery. 2009. 55 p. Dissertation (Masters in Computer Science Engineering) - International Institute of Information Technology, Hyderabad India. 2009.

Haralick, R.; Shapiro, L. **Computer and Robot Vision vol I**. Boston: Addison-Wesley, 1992, 630 p.