Road Surface Modeling from Vehicle-Borne Point Cloud by Profile Analysis

Chih-Wen, Wu[1] and Shih-Hong, Chio[2]

[1]Graduate student, Dept. of Land Economics, National Chengchi University,

NO.64, Sec.2, ZhiNan Rd., Wenshan District, Taipei City 11605, Taiwan; Tel: +886-2-29393091#51657;

E-mail: 97207414@nccu.edu.tw

[2]Associate Professor, Dept. of Land Economics, National Chengchi University,

NO.64, Sec.2, ZhiNan Rd., Wenshan District, Taipei City 11605, Taiwan; Tel: +886-2-29393091#51657;

E-mail: chio0119@nccu.edu.tw

KEY WORDS: vehicle-borne LiDAR system, point cloud, road surface model, feature constraints

ABSTRACT: With the promotion of the accuracy of the position and orientation system (POS), mobile mapping system (MMS) is developed quickly. Especially, vehicle-borne LiDAR system combines the technology of the POS and LiDAR (light detection and ranging) to collect precise 3D point cloud of the detailed road corridor efficiently. However, the processing of the plenty of those point could is time-consuming. Therefore, those point cloud have to be preprocessed for post-processing more efficiently. Since the road surface is the most important information in the road corridor, this study will focus on the road surface modeling. First of all, the collected 3D point cloud will be organized and projected onto vertical profiles which are defined by the moving trajectory of vehicle-borne LiDAR system. Then, 3D projected points in each vertical profile will be further segmented into line segments by Douglas–Peucker algorithm. Next, the scene knowledge will be employed to extract line segment point on road surface. Finally, the boundaries of the road surface will be fitted by roubust estimation from the points on the road boundary extracted from road line segment points. From the results, it shows the feasibility of the proposed algorithm.

1. INTRODUCTION

Recently, the terrestrial LiDAR (Light Detection And Ranging) can collect precise and high-density 3D point cloud of the terrain objects and terrain surface. With the improvement of the accuracy of the POS (Position and Orientation System), it can be installed on both vehicle-borne and airborne LiDAR system to capture spatial information more efficiently. Because of the occlusion problem, the detailed road information cannot be collected by the airborne LiDAR system. Therefore, in this study the 3-D point cloud collected by vehicle borne LiDAR system, one kind of the MMS (mobile mapping system), is employed to reconstruct the road surface model. However, the processing of the 3-D point data is time-consuming. Therefore, the collected 3-D point cloud needs to be structured or organized e.g. Octree-structure (Wang and Tseng, 2004) and scan lines (Lehtomäki et al., 2010), in order to reduce the processing time. In this study, point cloud will be structured and projected onto many vertical profiles defined by the moving trajectory of vehicle-borne LiDAR system. After the point cloud is structured, the projected 3-D points on each vertical profile will be further segmented into line segments by Douglas–Peucker algorithm. Then, the segmented line points on road surface will be extracted by using the scene knowledge, similar to the concepts of feature constraint (Pu and Vosselman, 2006). Finally, the road boundaries will be fitted by robust estimation from the boundary points extracted from the road line segment points.

2. METHODOLOGY

In this section, the steps to model the road surface from vehicle-borne point cloud by profile analysis will be described.

2.1. The Structuralization of Point Cloud

Since the processing of vehicle-borne point cloud data at one time is pretty time-consuming, those point cloud will be structured according to sequent vertical profiles. Those sequent vertical profiles are defined as the planes that are vertical to the x-y plane, i.e. horizontal ground, and along the moving trajectory of the vehicle. The 3-D points between two sequent vertical profiles are organized into the same group. Because 3-D points in the same group will be projected onto the vertical profiles for the subsequent processing, those vertical profiles are also called as projection planes in this study. For projecting the 3-D points in the same group onto the projection plane, the related equation should be formulated. As shown in Figure 2.1, Projection Plane(i) is vertical to the horizontal ground and perpendicular to the Direction(i). Therefore, the equation of Projection Plane(i) is defined by equation 2.1.

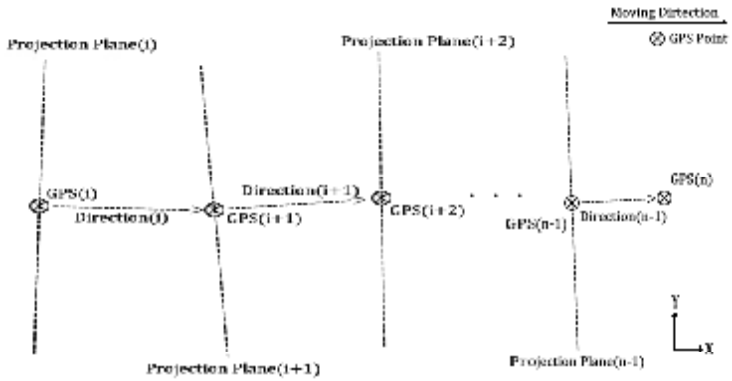


Figure 2.1: the relationship between GPS trajectory and projection plane

$$\mathrm{GPS(i)} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}$$

$$\mathrm{Direction(i)} = \begin{bmatrix} x_{i+1} - x_i \\ y_{i+1} - y_i \\ z_{i+1} - z_i \end{bmatrix}$$

$$\mathrm{ax + by + cz + d = 0}$$

$$a = (x_{i+1} - x_i)$$

$$b = (y_{i+1} - y_i)$$

$$c = 0$$

$$d = -(a \times x_i + \mathrm{b} \times y_i)$$

Equation 2.1:
the equation of Projection Plane(i)

After defining the projection plane, the vertical distance of every LiDAR points will be calculated the vertical distance to the projection plane such as Figure2.2. If the distance is lower than the threshold, the point will project to the plane such as Fifgure2.3.

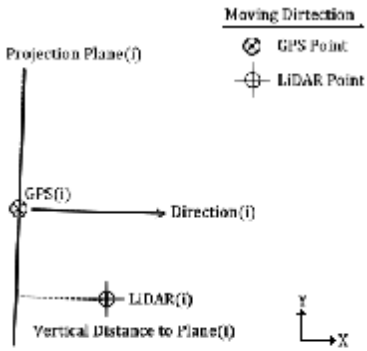


Figure 2.2: the relationship between LiDAR point and projection plane

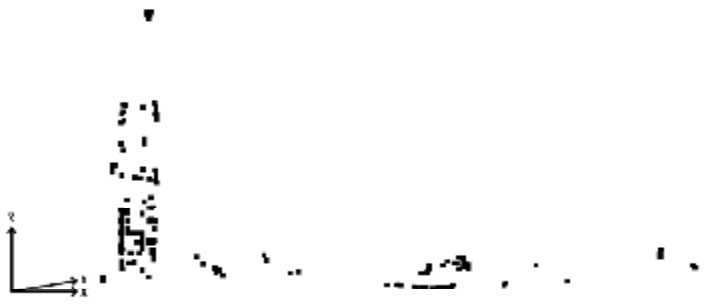


Figure 2.3: the result of projecting to the project plane

2.2. Segmented into Line Segment Points

After the preprocessing described in section 2.1, the projected points on each projection plane will be segmented into line segment points by Douglas–Peucker algorithm. The concept of Douglas–Peucker algorithm is shown as Figure 2.4. The projected points in the projection plane are shown as Figure 2.4(a). Firstly, the algorithm will find the point 1 and the farthest point 12. Then, the point with the longest distance to the line connected by points 1 and

12 will be found, in this case point 4 is found. If the longest distance exceeds one threshold, the projected points will be segmented into two groups. In Figure 2.4(a), the points will segment into two groups I: (points 1 to 4) and group II: (points 4 to 12) by the point 4. After the first segmentation, point group I (points 1 to 4) will be decided whether it is necessary to be segmented or not. In this case, if the distances of points 2 and 3 to the line connected by points 1 and 4 do not exceed the threshold, the point group will not be segmented. As to the point group II (points 4 to 12), the point 6 with the longest distance exceeds the threshold, the algorithm will continue to segment this point group into two groups, i.e. points 4 to 6 and points 6 to 12, as shown in Figure 2.4(b). The segmentation processing, see Figures 2.4 (c) and (d), will be repeated until no distance of points exceeds the threshold. The final segmentation, in this case, is shown as Figure 2.4(d), and all the segmentation location, called nodes, in each segment will be recorded by their projected coordinates. Each line segment is formed by a group projected points, which are called line segment points in this study.

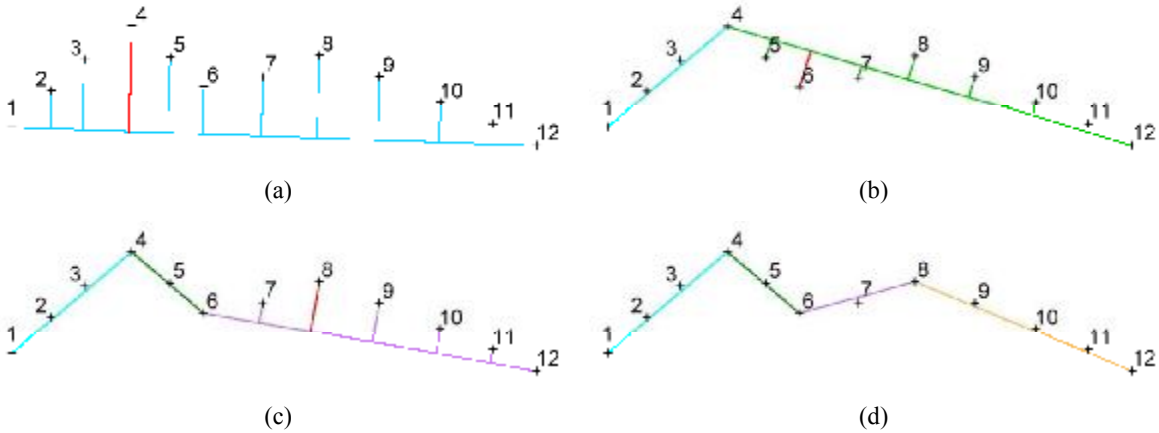


Figure 2.4: the concept of Douglas–Peucker algorithm

2.3. Classification of Line Segment Points

In this procedure, the scene knowledge will be used to classify each line segment points recorded by their projected point group into horizontal points, vertical points, and others. The scene knowledge is that most of 3-D points on the road corridor come from the man-made objects, e.g. road surface, building walls, the poles, etc, therefore, after the processing described in section 2.2, most of the line segments should be vertical or horizontal. The approach of classification is illustrated as shown in Figures 2.5 and 2.6. In Figures 2.5 and 2.6, points 1 to 5 are in same line segment and its corresponding vector is determined by the nodes, points 1 and 5, in this line segment. If the angle between the vector of line segment and vertical vector is lower than a threshold, this line segment points will be classified into vertical points, such as Figure 2.5. In other hand, if the angle between the vector and horizontal vector is lower than a threshold, this line segment points will be classified into horizontal points, see Figure 2.6. If the line segment points cannot be classified into the vertical points or horizontal points, those line segment points are labeled as others.

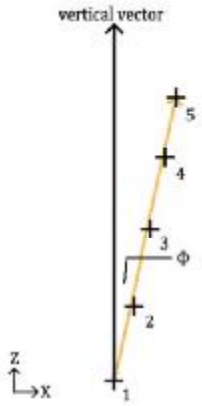


Figure 2.5: the vertical points

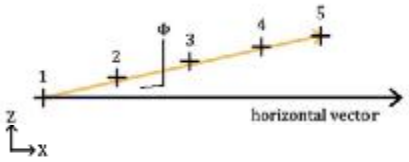


Figure 2.6: the horizontal points

2.4. Extraction of Road Segment Points

After the horizontal points are extracted, the subsequent step is to determine if the horizontal points belong to the road or not. In this step, two kinds of scene knowledge are used to extract the road segment points. One is the road elevation should be similar to the elevation of the moving trajectory minus the height of the GPS antenna above the road surface. The second one is that the road points should be located on the specific range relative to the moving trajectory of the vehicle. Figure 2.7 illustrates these two kinds of scene knowledge, where the GPS(i) means the moving trajectory of the vehicle at the projection plane i, and line segment CD represents the road surface. Based on the scene knowledge shown in Figure 2.7, road segment points have to satisfy two conditions. One is the elevation of the node, e.g. the nodes of segment CD, will be lower than the elevation of GPS(i) elevation minus the height between GPS(i) and road surface. The other is the horizontal distance between GPS(i) and the node, the nodes of segment CD in Figure 2.7, will be less than the distance between GPS(i) point and the road boundary. However, the approach might be in trouble such as Segment EF, a road traffic island, shown in Figure 2.7.

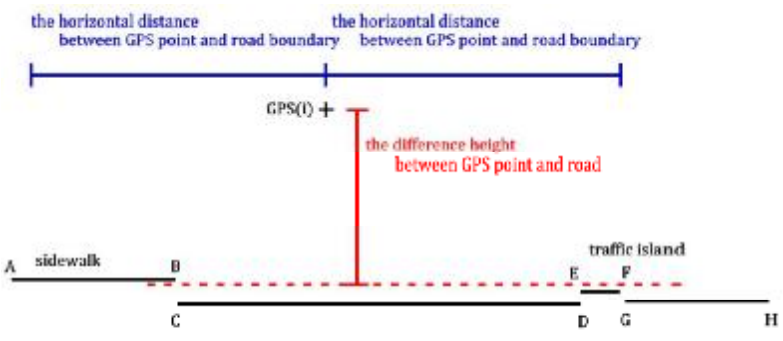


Figure 2.7: the scene knowledge used to extract the road segment points

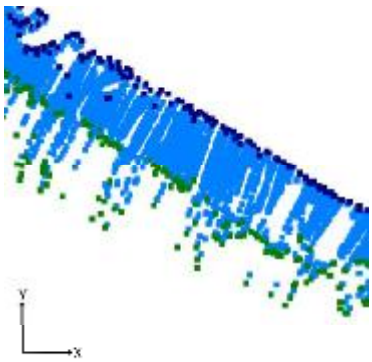


Figure 2.8: the nodes of road boundaries

2.5. Fitting the Road Boundary

After the extraction of road segment points, the next step is to determine the road boundary by least squares fitting. In this step, the road boundary is assumed as straight line. Firstly, the approach is to acquire the nodes on the road boundaries, i.e. the dark blue and dark green points in Figure 2.8.

Obviously, some wrong boundary points are extracted. Robust estimation (Kubik and Merchant,1988) is used to fit

the road boundary. According to the theory of robust estimation, the weights of the wrong observations will decrease when the residuals of the observations increase based on the formula (1). Thus, the wrong observations would not affect the fitting results.

$$p^{(v+1)} = p^{(v)} \times f(v)$$

$$f(v) = \begin{cases} 1\,, & \frac{|v|\sqrt{p_0}}{m_0} < c \\ \exp\left(\frac{-|v|\sqrt{p_0}}{m_0}\right), & \frac{|v|\sqrt{p_0}}{m_0} \geq c \end{cases}, c = 3 \quad (1)$$

p(v): weight function f(v): residual function v: residual of observation c: constant=3

P0： coefficient of weight m0: standard deviation of observation

The road boundary is regarded as the straight line. By using the (x,y) coordinates, the observation equation can be expressed by equation (2) and the observation equation can be also expressed by matrix method like equation (3).

$y_i + v_i = a x_i + b$ (2)

x_i, y_i : x and y coordinates of point a, b : the parameter of the line v_i: residuals of y coordinates of point

$AX = L + V$ (3)

$$A = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_{n-1} & 1 \\ x_n & 1 \end{bmatrix} \quad X = \begin{bmatrix} a \\ b \end{bmatrix} \quad L = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix} \quad V = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{n-1} \\ v_n \end{bmatrix} \quad W = \begin{bmatrix} p_1 & 0 & \dots & 0 & 0 \\ 0 & p_2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & p_1 & 0 \\ 0 & 0 & \dots & 0 & p_1 \end{bmatrix}$$

According to least squares principle, the solution will be derived as $X=N^{-1}U$, where $N=A^TWA$, and $U = A^TWL$. Then, $X=N^{-1}U$ can be applied to determine a and b after iteration. Every iterations, the weights of each observation will be renewed based on equation (1).

3. TEST SITE AND DATA

The study area is near the TaiChung city hall in central Taiwan. The vehicle borne LiDAR points are used only from one scanner of RIEGL VMX-250 and the data of original point cloud is about 10 Gigabytes. Because the processing of the raw data is time-consuming, the test data is resampled and reduced to about a tenth part by choosing one point in every ten points randomly.

After the original data is reduced, as mentioned in section 2, the moving trajectory of the vehicle is used to structured and organized point cloud into subsequent profiles. According to the density of the point cloud, the width between the subsequent profiles is set to 5 cm in this study. Then, the approaches described from sections 2.2 to 2.5 are used for profile analysis for modeling road surface, in this study the road boundary is extracted for modeling the road surface.

Before modeling, five parameters are set in this study. First parameter is the threshold of Douglas–Peucker algorithm and this threshold is set as 10 cm based on the scene knowledge, i.e. the height difference between the sidewalk and the road surface. The second parameter is the angle threshold for the classification of line segment points and 10 degrees is set by try and error. The third and fourth parameters are the thresholds for the height between GPS point and road surface, and the horizontal distance between GPS point and the boundary of the road

to extract the road segment points. According to the scene knowledge, the height threshold is 1.795 m by the manual identification and the horizontal distance is 10 m based on the road average width in the test site. The last parameter is the convergent condition for robust fitting the road boundaries. This parameter is set as try and error. All of parameters are shown in Table 1.

Table1: All parameters setting

The thickness of each profile	5 (cm)
The threshold of Douglas–Peucker algorithm	10 (cm)
The angle threshold	10°
The height threshold	1.795 (m)
The horizontal distance threshold	10 (m)
The convergent condition for robust fitting	20 (cm)

4. RESULTS

The test aims at extracting the boundaries of 16 roads in test area. Due to the straight line assumption of the road boundaries, therefore, the curved road boundaries will not be discussed in this study. Figures 4.1 and 4.2 show the results of the road boundary points after the processing described from sections 2.1 to 2.4. From the results, most of the road boundary points are extracted.

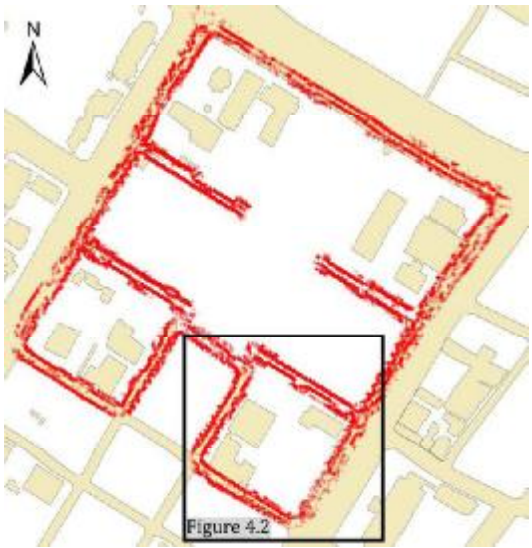
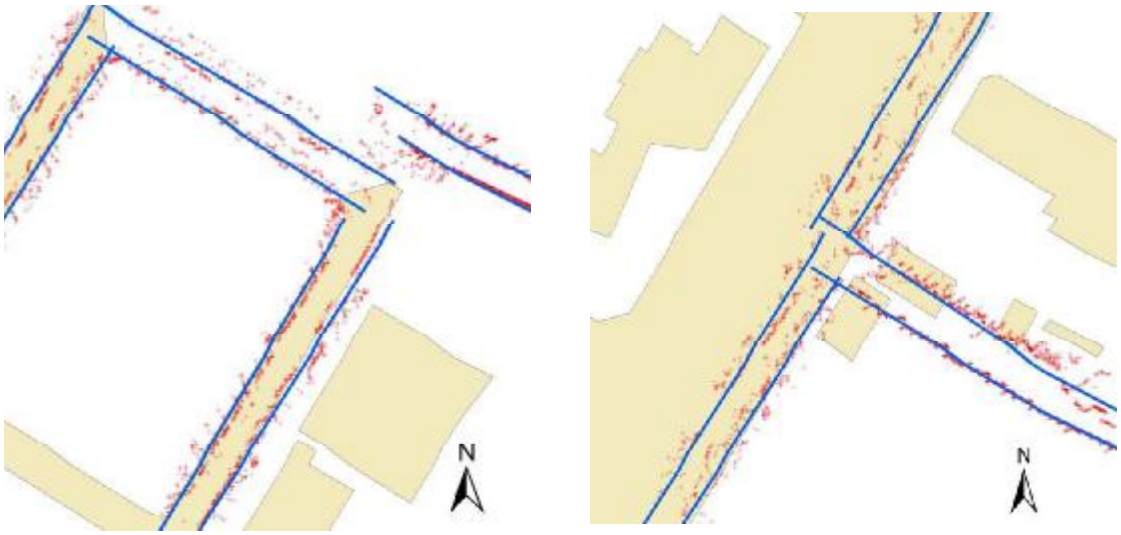


Figure 4.1: the result of the road boundary points

Figure 4.2: local zoom in the black block of Figure 4.1

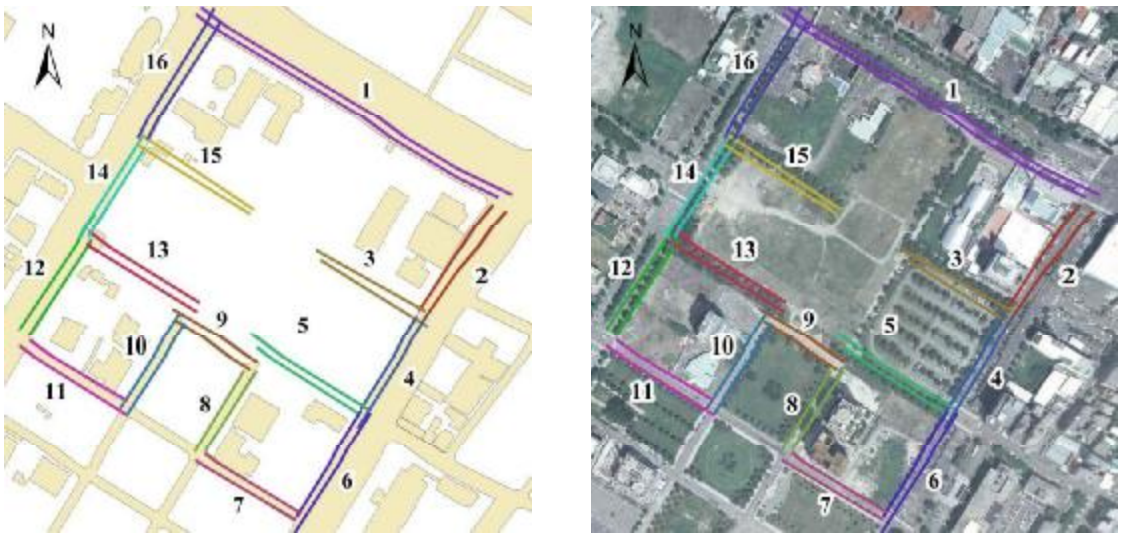
As shown in Figure 4.3, although most of the road boundary points are extracted, some wrong points are also extracted and some boundary points are missing. Therefore, robust estimation is utilized to fit the road boundaries. Figure 4.3 shows parts of the fitting results.



(a) (b)

Figure 4.3: the road boundary fitting by robust estimation

Finally, the results are compared with 1/5000 orthoimages and topographic maps, as shown in Figure 4.4. The detailed comparison to 1/5000 orthoimges and topographic maps for each road are shown in Figures 4.5 and 4.6.



(a) the result overlay with topographic maps (b) the result overlay with 1/5000 orthoimages

Figure 4.4: the qualitative evaluation is of extracted road boundary by overlay analysis

Because the road boundaries are assumed by the elevation change, therefore walksides do not belong to the road surface, the offset between road boundary and sidewalk is shown in Figure 4.5 (a). Additionally, the Road 1, 2, 4, 6, 12, 14 and 16 are road with multilane. Therefore, one-side of road boundary in the Road 1, 2, 4, 6, 12, 14 and 16 do not match to the topographic maps and 1/5000 orthoimages, see Figures 4.5 and 4.6. Because no information of Road 3, 5, 9, 13 and 15 on the topographic map and orthoimages, these 5 roads cannot be verified by overlay analysis. Any two straight line boundaries on road should be parallel in most roads, but some fitting boundaries are not really parallel. However, the result of Roads 7, 8, 10, and 11, as shown in Figure 4.5 (d), (e), and (f), are much more accurate than the other fitting results of road boundary and these road boundaries match the topographic maps well by visual analysis.

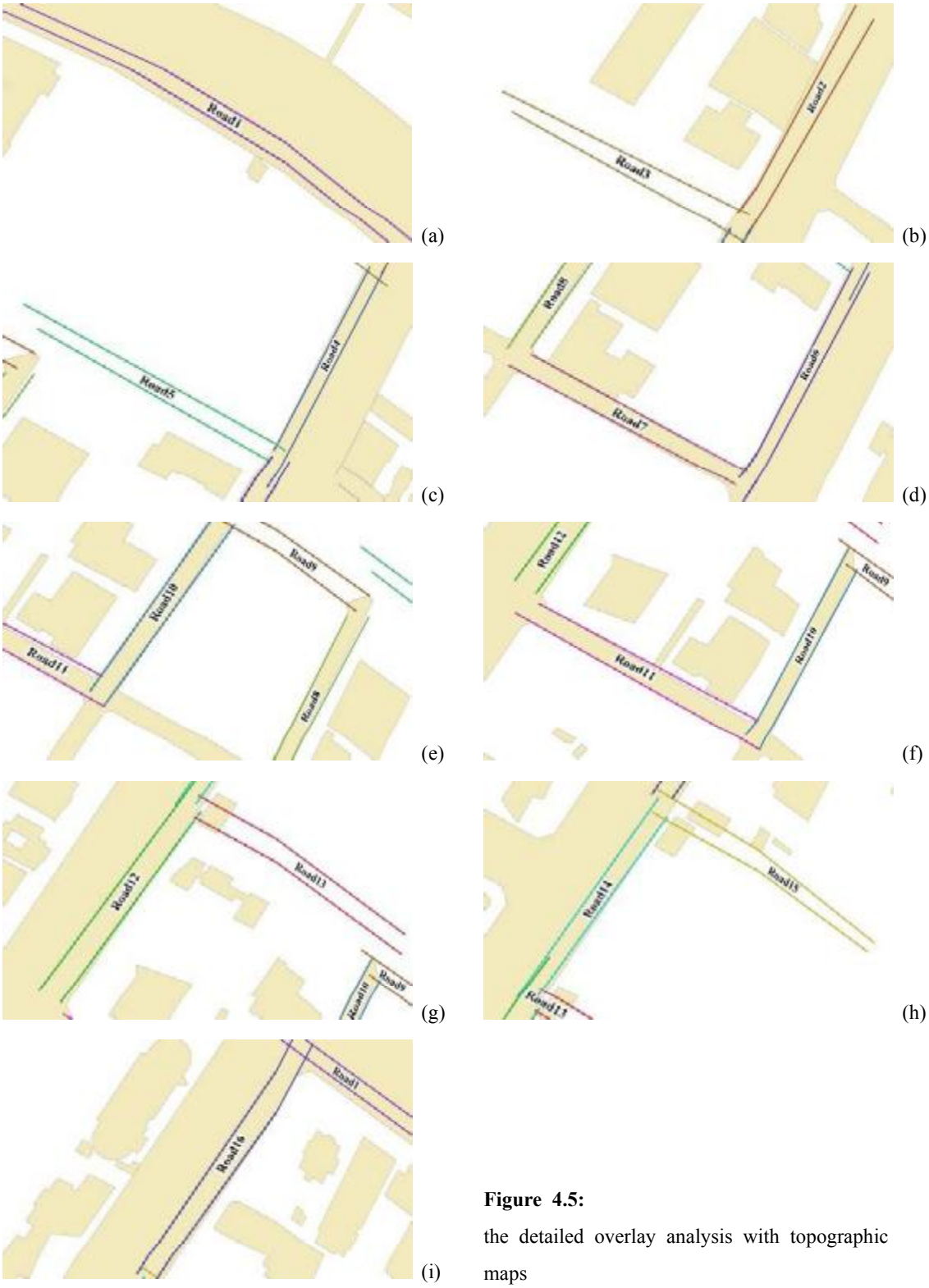


Figure 4.5:
the detailed overlay analysis with topographic maps

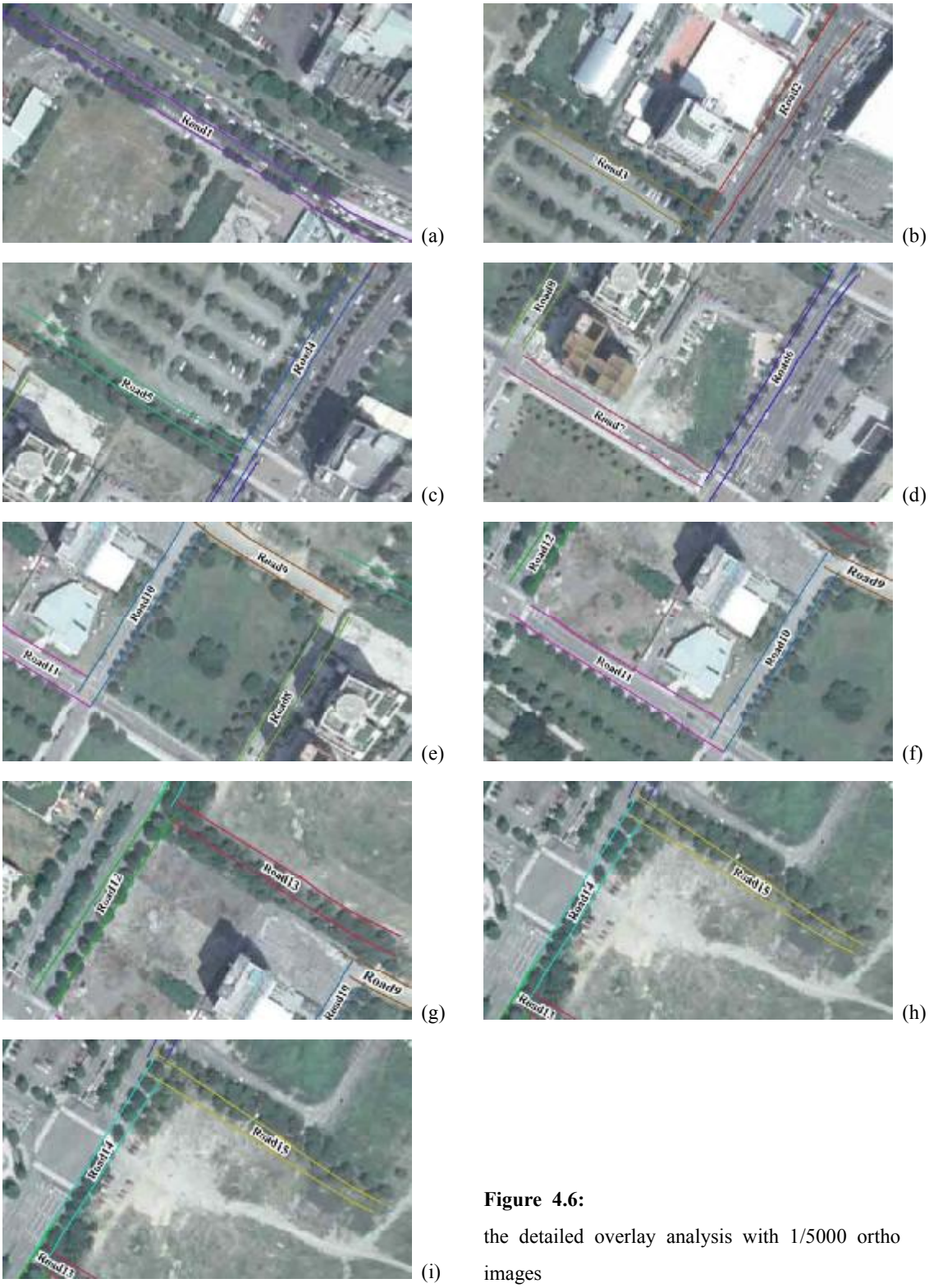


Figure 4.6:
the detailed overlay analysis with 1/5000 ortho images

5. CONLUSIONS & RECOMMODATIONS

Based on the results, it proves the feasibility of the proposed algorithm. Some factors will affect the results. The first one is the density of the point cloud. Because vehicle-borne LiDAR system cannot acquire accurate boundaries of the object, the result will be influenced by the density of the point cloud. If the data was reduced too much, the information of the boundaries will be lost. Except for the density, the other factor is the occlusion problem. In this case, cars near the road might block the road boundaries and influences the result. Although this study will fit wrong boundary points with low weights during robust estimation, the fitting quality is still worse when most of the road boundaries are occluded. However, in this study, robust estimation can reduce most influence from wrong points. Finally, the parameters setting will influence the result. For example, if the parameter setting is not suitable, the extraction of the road boundary will have many errors and the robust estimation cannot improve it.

Except for the above-mentioned conclusions, there are three recommendations from this study. Firstly, the elevation of road boundaries is not considered while fitting the road boundaries in this study. Therefore, by considering the elevation of road boundaries it will be helpful for the reconstruction of 3-D road model. Secondly, much more sense knowledge are taken into consideration, it will be also helpful for classification and might improve the classification result. Thirdly, the quantitative analysis should be done in the future.

REFERENCES:

Kubik, K. and D. Merchant, 1988. Photogrammetric Work Without Blunders. Photogrammetric Engineering and Remote Sensing, Vol. 54, No. 1, pp. 51-54.

Lehtomäki, M. , Jaakkola, A. , Hyyppä, J. , Kukko, A. and Kaartinen, H. , 2010 ,Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data , Remote Sensing , Volume: 2 - Issue: 3, 641-664.

Miao Wang and Yi-hsing Tseng, 2004, Lidar data segmentation and classification based on octree structure, ISPRS

Pu, S. and Vosselman, G. , 2006 , Automatic extraction of building features form terrestrial laser scanning , International Archives of photogrammetry, Remote Sensing and Spatial Information Sciences.