AUTOMATIC CLASSIFICATION OF LIDAR DATA INTO GROUND AND NON-GROUND POINTS

Yu-Chuan Chang[a], Ayman F. Habib[a], Dong Cheon Lee[b], and Jae-Hong Yom[b]

[a]Department of Geomatics Engineering, University of Calgary, Calgary, 2500 University Drive NW, Alberta, Canada T2N 1N4 - (ycchang, habib)@geomatics.ucalgary.ca
[b]Department of Geo-Informatics, Sejong University, Seoul, South Korea - (dclee, jhyom)@sejong.ac.kr

Commission: WG IV/3

KEY WORDS: LiDAR, DEM/DTM Extraction, Photogrammetry, Classification, Laser Scanning, Point Cloud

ABSTRACT:

Recently, automatic object extraction from Light Detection And Ranging (LiDAR) data has attracted great attention. The level of detail and the quality of the collected point cloud motivated the research community to investigate the possibility of automatic object extraction from such data. Prior accurate knowledge of terrain information is usually essential for the data to be usable in further processing, such as feature extraction, and to obtain better object detection results. In this paper, a new strategy for automatic terrain extraction from LiDAR data is presented. The proposed strategy is based on the fact that sudden elevation changes, which usually correspond to non-ground objects, will cause relief displacements in perspective views. The introduced relief displacements will occlude neighboring ground points. A Digital Surface Model (DSM) is first generated by resampling the irregular LiDAR point clouds to a regular grid. By using synthesized projection centers located above the DSM and analyzing the visibility maps in perspective images, we can classify the DSM into non-ground and ground hypotheses. Surface roughness and inherent noise in the point cloud will lead to some false hypotheses. By using a novel algorithm which combines plane fitting and statistical filtering to remove these false hypotheses, non-ground and ground points can be separated. The algorithm has been tested using both simulated and real datasets. The results have demonstrated that our approach can perform well with highly complex data from an urban area. In a comparison with the results obtained with TerraScan software, our algorithm showed the capability of producing better results while being less sensitive to used parameters.

1. INTRODUCTION

LiDAR technology has been demonstrated in recent years to be a prominent technique for the acquisition of highly dense and accurate information for physical surfaces. As LiDAR is a non-selective mapping method, the acquired data consists of a point cloud that includes bare-ground and non-ground objects such as trees and buildings. Methods of removing non-ground points, also referred to as filtering techniques, have been the focus of many researchers. Many applications, for example, the generation of contour lines for topographic maps, road engineering projects, and the delineation of flooding zones, among others, require the generation of a DTM from the ground points. A DTM can be produced by resampling those extracted ground points from LiDAR data. The filtering step is also essential for the data to be usable in further processing, such as in feature extraction. Building detection and reconstruction procedures for the generation of 3D city models can be facilitated by first detecting the non-ground points. The feature extraction and modeling procedures are also beneficial to applications such as change detection and database updating.

To satisfy the needs of these applications, the research community has been developing several techniques for the classification of LiDAR data. The first group of methods that can be identified in the literature are based on mathematical morphology. A method related to the erosion operator was proposed by Vosselman (2000). In this method, the acceptable height difference between two points is explicitly defined as a function of the distance between the points. Morphological filters have some drawbacks when certain features, such as large buildings and dense forest canopy, are involved. In such cases, a window size that is too small could be including only building

points, thereby classifying them as ground. However, a larger window size can potentially chop off hills that have a significant slope. Strategies such as the use of multiple window sizes, as proposed by Kilian et al. (1996), and the one developed by Zhang et al. (2003), which gradually increases the window size, might help in overcoming these problems. However, the success of these types of filters is strongly dependent on the selection of the discriminant function parameters. The second group of filters are based on the progressive densification of a TIN (Triangulated Irregular Network). In Axelsson (2000), ground points are classified by iteratively building a triangulated surface model. The third group of methods are based on linear prediction and hierarchic robust interpolation (Kraus and Pfeifer, 2001). The approach is based on a surface model defined for the entire point set that iteratively approaches the ground surface. However, these two groups of methodologies cannot handle the surface with low and complex objects very well, as reported by Sithole and Vosselman (2004).

Approaches that rely on segmentation are also found in the literature. Jacobsen and Lohmann (2003) developed a method that first segments the data and then classifies the segments as either ground segments or off-terrain segments, based on neighborhood height differences. When dealing with large areas, segmentation techniques require expensive computation for processing. Other filtering algorithms are also described by Elmqvist et al. (2001), and Brovelli et al. (2002), among others. A detailed comparison of some filters is provided in Sithole and Vosselman (2004). The experimental study conducted shows that in flat and uncomplicated landscapes, all the algorithms give satisfactory results. However, significant differences in the accuracy of these methods appear when landscapes containing

steep slopes and discontinuities. These differences are the result of the differing abilities of the algorithms to preserve discontinuities while detecting large and low objects.

In this paper, a new approach for the automatic extraction of terrain points from LiDAR data is presented. The next section will briefly describe our approach. This discussion will be followed by the proposed methodology for extracting non-ground points. The Experimental Results section presents the descriptions of our datasets and results. Finally, concluding remarks regarding the performance of the proposed technique, together with future research directions are summarized.

2. METHODOLOGY

In a perspective image, we can see relief displacement caused by the height of the corresponding object point above or below the datum. Relief displacement is directly proportional to the radial distance and the object height above the datum. However, relief displacement is inversely proportional to the flying height above the datum. A larger radial distance and a perspective center with a lower height can cause more occlusions in the image. The concept of our new approach is based on occlusion detection. Non-ground points can cause occlusions in perspective views. Therefore, if occlusions can be detected, and we can find out which points are causing the occlusions, then these points would be identified as non-ground points. In our approach, we generate a DSM grid from irregular LiDAR point cloud. Using this DSM, once the occlusions are detected using synthesized perspective centers, the points producing the occlusions are identified. After removing the effects of the roughness of the terrain, non-ground points and ground points can be separated from one another. Figure 1 summarizes the procedure.
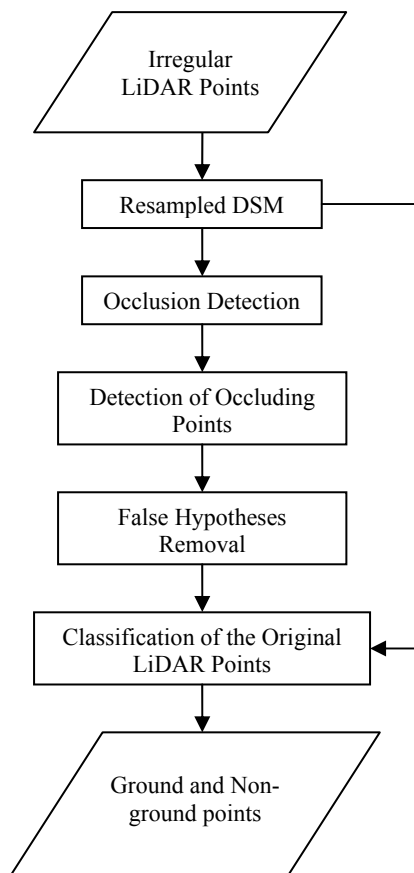
Figure 1. Flowchart of LiDAR data classification.

2.1. DSM Generation

A LiDAR point cloud is obtained as an irregularly spaced set of points. For most analytical processes, processing this irregular data format is time-consuming, and converting the points to a regular grid for analysis and visualization increases the efficiency. The pixel size has to be determined before resampling. Reducing information loss is important, as is keeping the redundancy at a minimum, while resampling. A very large ground sampling distance (GSD) for the resampled DSM will increase the information loss. However, the redundancy increases, as do the storage requirements, if the GSD is very small. To satisfy these requirements, the optimum GSD for resampling can be estimated to be equal to the average point density of the LiDAR data. In order to keep the edges from being blurred by some low pass filters, we use the nearest neighbor method for resampling. The elevation of each grid point is assigned the elevation of the closest original LiDAR point. If there is more than one point located in a pixel, we pick the one with the lowest height and assign its height to the pixel.

2.2. Identification of the Points Causing Occlusion

In this paper, the off-nadir angle to the line of sight will be denoted as the α angle, as in Figure 2. As we move away from the nadir point, the off-nadir angle α is supposed to increase (Habib et al., 2007). As long as the α angle increases while moving away from the nadir point, the DSM cells along the radial direction will be visible in the image in question. For example, points A and B are visible in Fig. 2 since their corresponding off-nadir angles increase as we move away from the nadir point. Occlusions, on the other hand, can be detected whenever there is an apparent decrease in the off-nadir angle α while proceeding away from the nadir point. This occlusion will persist until the off-nadir angle α exceeds the angle associated with the last visible point. In Fig. 2, we find that because $\alpha_D < \alpha_C$, an occlusion is detected at point D. This occlusion will persist until point E, at which $\alpha_E > \alpha_C$. After an occlusion has been detected, the points causing the occlusion can be identified while tracing a path toward the nadir point. Figure 2 shows how the points causing an occlusion can be determined using the triangle composed of the last visible point, the first occluded point, and the perspective center. The non-ground points can be traced until the off-nadir angle α is equal to the angle associated with the first occluded point. In Figure 2, an occlusion is detected because $\alpha_D < \alpha_C$. Point C is defined as the last visible point for this search, and point D is taken to be the first occluded point. A backward search for points causing occlusions can then be carried out. Those points with off-nadir angles larger than α_D are defined as points causing occlusion. For example, point B is taken to be a non-ground point since $\alpha_B > \alpha_D$. The backward tracing stops when we find an off-nadir angle smaller than α_D. In Figure 2, the tracing stops at point A because $\alpha_A < \alpha_D$.

In order to obtain a more complete list of the points causing occlusions, we need to enhance the procedure's capability of detecting these points. Adjusting the locations of the synthesized projection centers relative to the DSM can maximize the introduced occlusions. If the elevation of the perspective center can be adjusted to be as close as possible to the height of the non-ground object, then the capability of detecting the points causing occlusions can be improved. A

larger radial distances between a DSM cell and the nadir point can be also helpful in detecting the points producing occlusions. By combining both of the above methods, we can enhance the capability of our procedure to detect the points causing occlusions.
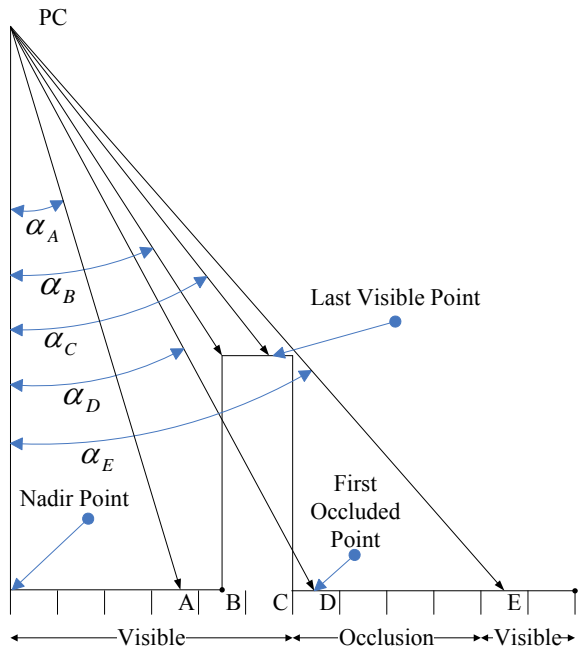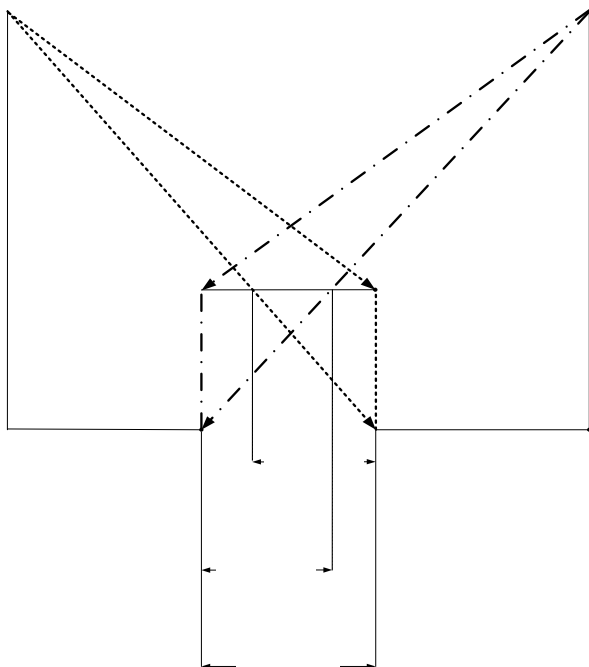

Figure 2. Occlusion detection in perspective views.

Figure 3: Two perspective centers on opposite sides used to detect all the non-ground points

When dealing with large buildings, detecting non-ground points using only one perspective center could be a challenge. In

Figure 3, the building requires two perspective centers on opposite sides in order to detect all the non-ground points for each vertical profile. It is necessary to check all possible occluding directions. Considering that every pixel has eight possible neighbors that could produce occlusions, for each pixel, we use 8 perspective centers with heights close to the maximum elevation of the entire area; this way, the points causing occlusions can be detected more thoroughly. For the same reason, larger radial distances between DSM cells and the nadir points are also required. The locations of the perspective centers are outside the region of interest, at a distance d.

The algorithm is tested using the artificial data with sloping terrain shown in Figure 4. Some objects are located above the ground, and some noise is added to the DSM. Using the proposed algorithm with synthesized perspective centers, potential non-ground points are separated from ground points, as shown in Figure 5, in which the white points are the points causing occlusions and the black points are the extracted ground points. The result of classification in Figure 5 includes some false hypotheses which are caused by noise in the terrain surface.
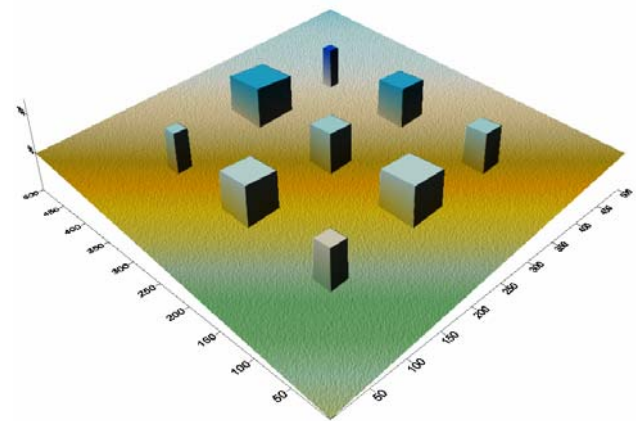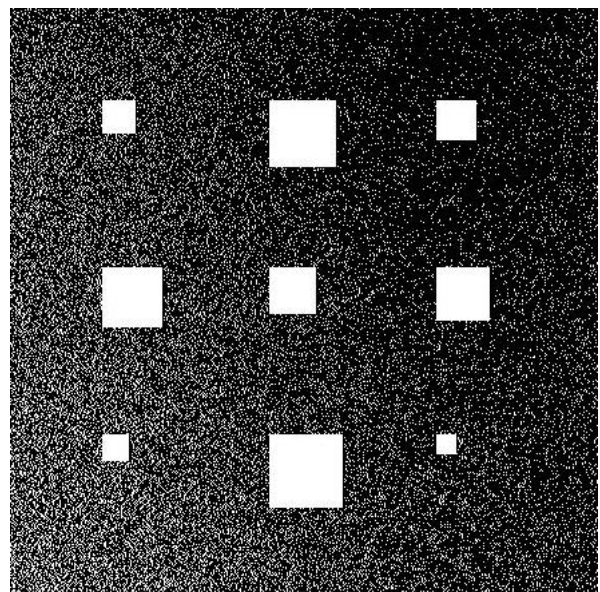

Figure 4: The simulated surface Model.
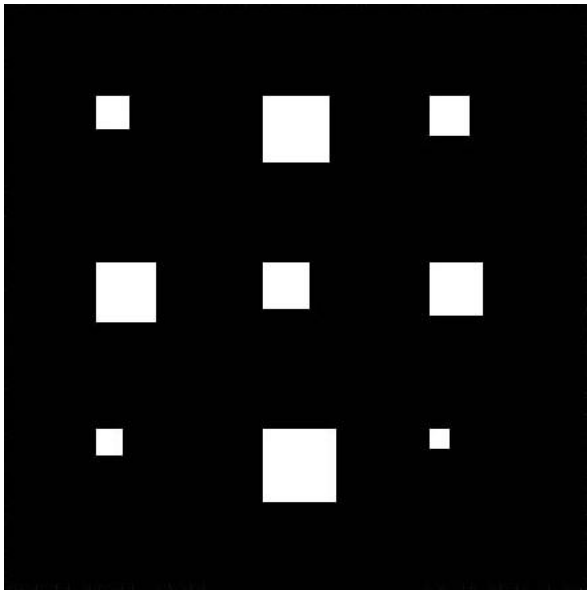

Figure 5: Points causing occlusions.

Figure 6: Detected non-ground points

2.3. Using a Statistical Filter to Remove the Effects of Surface Roughness

We want to use the potential non-ground points outputted from the previous procedure to extract non-ground points, so surface roughness and noise should be removed. When comparing the resulting potential ground points with potential non-ground points, the potential ground point results are more reliable, and thus would make a good reference set of points. The terrain can be considered a random field, in which the elevation can be approximated to be normally distributed with a mean μ and a variance σ^2.

In most cases, the surface of the terrain is supposed to be continuous; therefore, the distribution of terrain points is more suitable for being our reference than the distribution of non-ground points. In analyzing the histogram of the elevations of the detected ground points in a local area, we consider the points located beyond 2 STD from the mean of the distribution, where the probability of having a terrain point is only 2%, to be wrongly classified non-ground points that need to be corrected. These points are classified as ground points in the previous procedure because of the rough and uneven surfaces of the non-ground objects. We consider the points located within 1.5 STD from the mean of the distribution, where the probability of having a terrain point is 93%, to be reliable signals that should be kept. These points could have been identified as non-ground points because of the roughness of the ground surface.

The concept above can be implemented as a filter. Only the center of the filter window is examined, using the distribution of the neighboring ground pixels within the window. In order to have enough samples to generate a reliable distribution for re-classification, the filter window size should be adaptively increased if the number of potential terrain samples in this local neighborhood is less than a pre-defined number which is chosen to be 100 in our case. Through the moving window procedure, all the pixels can be checked. The statistical filter is used to remove the defects caused by the false hypotheses in the surface. The ground points extracted are shown in Figure 6.

When dealing with the terrain with very large slope angle, a standard deviation of a histogram could become very huge. In

order to handle the terrain with various slope angles, more constraints are needed to improve refinement of the classified points. Combining plane fitting with the statistical filter together (Fig. 7), a new method for correcting false hypotheses has been developed. Using potential ground points after occlusion detection in the local block as an input, the plane fitting procedure estimates the most probable plane which can be used to represent the terrain. The plane fitting procedure is performed through a least squares adjustment process by minimizing the summation of normal distances between the potential terrain points and the estimated plane. In order to determine where the higher probability of having a terrain points could happen, the standard deviation of normal distances between the estimated plane and the potential terrain points within the local block is first computed. Then using a multiple of the standard deviation, we create a buffer around the computed plane (Fig. 8 and Fig. 9). The central point of the local block is defined as non-ground if the point is located outside the buffer. Otherwise, the point is taken to be a ground point. If the estimation procedure for plane fitting cannot be convergent, a statistical filter can be used to correct false hypotheses.

A procedure for the classification of a regularly spaced surface model has been introduced above. After classifying the DSM into ground and non-ground pixels, we can classify the original LiDAR points based on their proximity to the classified DSM cells. Each cell, however, can contain more than one LiDAR point, and thus we must consider that only the lowest point within each cell was used in creating the DSM. If several LiDAR points lie in a DSM cell, which has been classified as a terrain point, then the lowest LiDAR point is classified as terrain. The classification of the remaining points depends on their height relative to the lowest LiDAR point. If the heights of the other points are significantly higher than the height of the lowest LiDAR point, these points are classified as non-ground. In the case that a cell is classified as non-ground, then all LiDAR points in this cell are classified as non-ground.
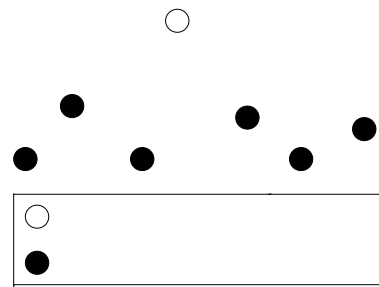

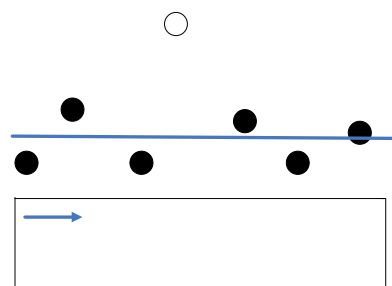Figure 7: Potential ground points and potential non-ground points in the local block

Figure 8: Plane fitting using potential ground points in the local block
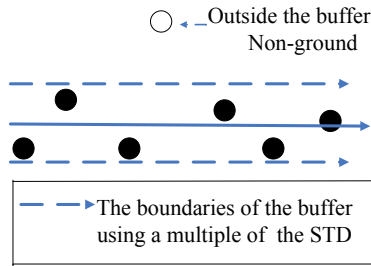
Figure 9: Identified outliners based on a buffer surrounding the estimated plane.

3. EXPERIMENTS

The proposed algorithm was tested using the set of artificial data shown in Figure 4, 5 and 6. For the simulated datasets, our results demonstrate 100% accurate classification of ground and non-ground points. The results have shown that this algorithm can handle the simulated sloping and hilly data effectively. This approach was also tested on real LiDAR data. In comparing our results with the ground truth, the number of misclassified points divided by the total number of points can give us the error rate, which, in this case, was calculated as 4.6896% (Chang et al., 2007). These results have demonstrated that our approach can perform well with highly complex and unpredictable data from an urban area.

We also compared our results with those produced using TerraScan. As shown in Figure 10(a), we chose an experimental area around the C-Train track near the University of Calgary. One can see a C-Train track extending into a tunnel under the ground in Figure 10(a). In cases like this, the default parameters of our algorithm are good enough to produce acceptable results. The parameters for ground and non-ground classification using TerraScan, on the other hand, need to be adjusted iteratively. After fine-tuning the parameters, we computed the best results from TerraScan and compared them with our results. Figures 11(a) and 11(b) show the extracted ground points and non-ground points using the proposed approach in this paper, while the extracted terrain point and non-ground points using TerraScan are shown in Figures 12(a) and 12(b).

The experimental results show that our algorithm can produce competitive results when compared with those obtained from TerraScan. In some areas, our approach can delivered better results. The default parameters of our algorithm can produce stable results in most cases; however, the parameters for the TerraScan function need to be adjusted iteratively for each case. Because the function of non-ground and ground point classification in the TerraScan software is designed mainly for DTM generation, the accuracy of the ground and non-ground classification is not so critical for the purpose of approximated DTM generation. Once enough ground points can be sampled, a DTM can be computed using an interpolation method.

4. CONCLUSION

This research presented a robust algorithm for the automated extraction of non-ground points from LiDAR point clouds by detecting points that produce occlusions. Following the occlusion detection, a statistical filter can be used to remove the effects of the terrain roughness and noise. Throughout the experiments, the proposed procedure separated the LiDAR non-

ground and ground points from one another successfully. The results have also shown that the algorithm performs effectively in simulated hilly terrain and in urban areas. In a comparison with the results obtained with the TerraScan software, our algorithm demonstrated the capability of producing more competitive outputs.

Future research will be extended to more complex scenes. In the next stage of research, non-ground points can also be classified into different objects such as buildings, trees, and cars, etc. Multi-return and intensity information will be taken into consideration in future work.
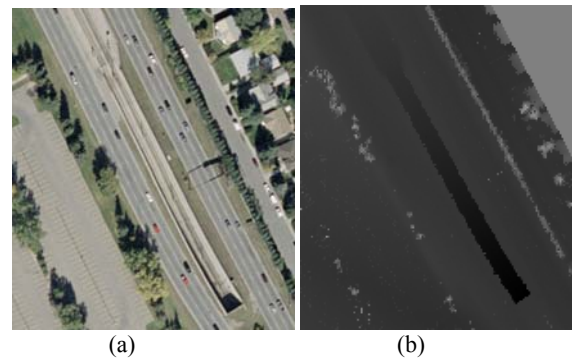

| (a) | (b) |

Figure 10: (a) The referenced aerial photo over the area covered by the LiDAR dataset. (b) The resampled DSM using LiDAR data.

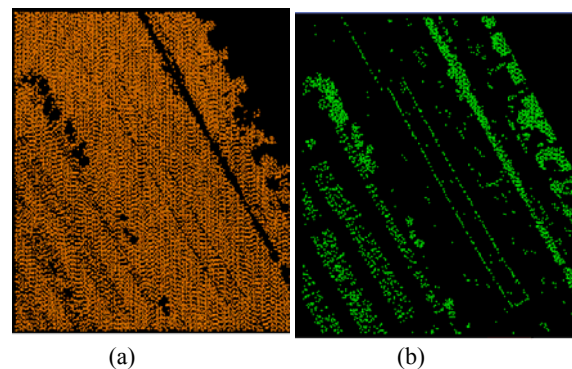| (a) | (b) |

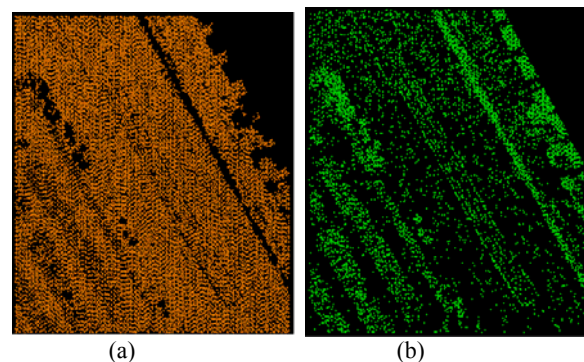Figure 11: (a). Ground points and (b). non-ground points extracted using the proposed method.

| (a) | (b) |

Figure 12: (a). Ground points and (b). non-ground points extracted using TerraScan.

ACKNOWLEDGEMENTS

REFERENCES

Axelsson, P., 2000. DEM generation from laser scanner data using adaptive TIN models. In: *The International Archives of the Photogrammetry and Remote Sensing*, 33 (B4/1), pp. 110–117.

Brovelli, M. A., Cannata, M. and Longoni, U.M., 2002. Managing and processing LiDAR data within GRASS. *Proc. GRASS Users Conference*, Trento, Italy, 11 – 13 September. University of Trento, Italy.

Chang, Y.C., Kim, C., Kersting, A.P. and Habib, A.F., 2007. New approach for dtm generation from LiDAR data, *The 28th Asian Conference on Remote Sensing (ACRS)*, Kuala Lumpur, Malaysia, November

Elmqvist, M., Jungert, E., Lantz, F., Persson, A., Soderman, U., 2001. Terrain modelling and analysis using laser scanner data. In: *The International Archives of the Photogrammetry and Remote Sensing*, 34 (3/W4), pp. 219– 227.

Habib, A., Kim, E. and Kim, C., 2007. New methodologies for true ortho-photo generation, *Photogrammetric Engineering and Remote Sensing*, 73(1), pp. 25– 36

Jacobsen, K. and Lohmann, P. 2003. Segmented filtering of laser scanner DSMs. In: *The International Archives of Photogrammetry and Remote Sensing*, 34 (3/W13).

Kilian, J., Haala, N., and Englich, M. 1996. Capture and evaluation of airborne laser scanner data. In: *The International Archives of Photogrammetry and Remote Sensing*, 31(B3), pp.383–388.

Kraus, K. and Pfeifer, N. 2001. Advanced DTM generation from LiDAR data. In: *The International Archives of Photogrammetry and Remote Sensing*, 34(3/W4) , pp. 23–30.

Sithole, G., Vosselman, G. 2004. Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59 (1-2), pp. 85–101.

Vosselman, G. 2000. Slope based filtering of laser altimetry data. In: *The International Archives of Photogrammetry and Remote Sensing*, 33(B3), pp. 935–942.

Zhang, K., Cheng, S.C., Whitman, D., Shyu, M.L., Yan, J. and Zhang, C. 2003. A progressive morphological filter for removing non-ground measurements from airborne LiDAR data. *IEEE Transactions on Geoscience and Remote Sensing*, 41(4), pp. 872–882.