SHORELINE EXTRACTION FROM THE INTEGRATION OF LIDAR POINT CLOUD DATA AND AERIAL ORTHOPHOTOS USING MEAN SHIFT SEGMENTATION

I-Chieh Lee, Bo Wu, and Ron Li
Mapping and GIS Laboratory
Dept. of Civil & Environmental Engineering & Geodetic Science, The Ohio State University
470 Hitchcock Hall, 2070 Neil Avenue, Columbus, OH 43210-1275
lee.3007@osu.edu

ABSTRACT

A method for shoreline extraction from integrated LiDAR point cloud data and aerial orthophotos is presented. First, a Mean Shift Algorithm is used for LiDAR point segmentation. The horizontal position and elevation of the LiDAR point plus color information obtained from the corresponding orthophoto are used as the point features in the Mean Shift Algorithm. Due to the homogenous nature of the elevation and color distribution of a water surface, LiDAR points distributed on the water surface and on the ground can be classified using Mean Shift Algorithm in a semi-supervised manner. Second, a modified convex hull algorithm is used to determine the boundary of the classified LiDAR points. The shoreline is defined as the result of the separation boundary between the LiDAR points belonging to water and those belonging to non-water. The experiment, which used LiDAR data and orthophotos acquired at the same time in Portsmouth, New Hampshire, shows that the accuracy of the derived shoreline is an improvement over LiDAR point spacing.

INTRODUCTION

A shoreline is defined as the line of contact between land and a water body. It is a constantly changing line due to the rise and fall of the water level (Li et al., 2002a). In the past, shoreline extraction has usually been done using photogrammetric methods (Li et al., 2002b; Woolard, 2003). These methods need plenty of ground surveys and operators to calculate the results. In recent decades, LiDAR systems have been introduced to generate shorelines (Slama et al., 1980; Ingham, 1992; Li et al., 2002a).The common method of extracting shorelines from a LiDAR point cloud is to generate a DSM, or profile, from the point cloud and then make an intersection with a water level, or water surface elevation (Robertson et al., 2004; Stockdon et al., 2002). Although this method is simple and effective, it has restrictions such as that the intersecting water level can only be higher than the water level in the LiDAR dataset. Otherwise, both topological and bathymetry LiDAR should be used to create the Coastal Terrain Model (CTM). In addition, there are offsets between LiDAR data strips and gauge station measurements, even if they are in the same vertical datum (e.g., NAVD 88). Translating LiDAR point elevation to a gravitational vertical datum using GCPs (ground control points) and strip adjustment must be done prior to the process. Another type of method uses segmentation-based image processing. Liu et al. (2007) first re-sampled the LiDAR point cloud into grid data and then used an image processing method that included region scanning, morphology operation, line tracing and vectorization to analyze the shoreline boundary. However, these methods will not keep the original LiDAR point cloud resolution.

This paper focuses on the extraction of shoreline by integrating LiDAR point cloud data with aerial orthophotos. We have developed a Mean Shift Algorithm to extract the instantaneous shoreline by classifying points as belonging to the water surface or to land, then finding the intersecting boundary between these classes. The result would be the instantaneous shoreline.

SHORELINE EXTRACTION PROCEDURE

LiDAR intensity has the potential to be used in LiDAR point cloud segmentation to identify the land LiDAR data. However, the LiDAR products over areas of water area differ from those in over land. LiDAR surveys in land areas can always obtain dense measured points with good intensity readings. However, in areas over water, only when the laser beam emits vertically into the water surface can the receiver obtain most of the photons reflected

back and thus obtain a maximum reading of the LiDAR intensity. When the scan angle of the laser beam is large, the reflecting beam is shooting away from the receiver, causing the signal not to be received. As a result, there will be no LiDAR point at all within that scan angle. Therefore, this paper did not use the LiDAR intensity for shoreline extraction. Instead, the 3-D position information and the corresponding color information were employed.

Input data used in our method is the LiDAR point cloud and the orthoimages that were acquired simultaneously. The procedure for generating an instantaneous shoreline is illustrated below (Figure 1). First, a learning procedure is performed to determine the parameters of the Mean Shift Algorithm. Second, color information (R,G,B) is assigned to every LiDAR point. Third, mean shift filtering is applied to the LiDAR elevation (Z) and color information (R, G, B) to minimize the segment numbers. Fourth, point density (PD) is calculated for every LiDAR point. Fifth, the LiDAR point information (X,Y,Z) and corresponding color information from the orthoimages (R,G,B) are used to segment the LiDAR points. Sixth, the parameters calculated from the learning procedure are used to group the segmented points into classes: only two classes are considered in this procedure, points on land and points on water. Finally, an instantaneous shoreline can be extracted by tracing the boundary of the land. The following sections give the details of this method.

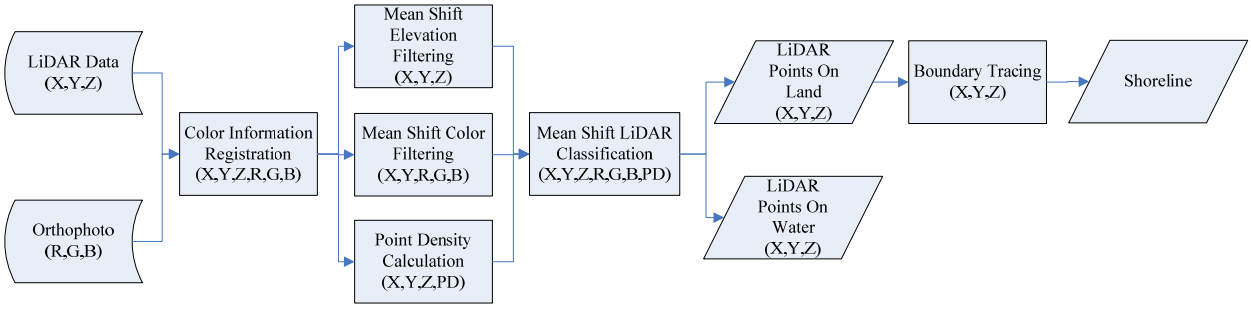

Figure 1. Flow chart of the shoreline extraction procedure.

Registration of LiDAR Data and Orthoimages

Before the color information can be used in our research, the orthoimages and the LiDAR data have to be registered. Since the aerial photographs and the LiDAR data are acquired simultaneously, the same set of GPS and INS measurements can be used in both aerial triangulation and determination of LiDAR coordinates. Consequently, we assume that the orthoimages and LiDAR are registered. If the aerial photographs and the LiDAR data are not taken simultaneously, then a precise registration should be done. In order to assign color information for each LiDAR point, we project the LiDAR points onto the orthoimages and then the color information (R, G, and B components) can be derived from the orthoimages and assigned to each of the LiDAR points. For the data sets used in our experiment, the registration between the LiDAR and orthoimage was done by National Geodetic Survey (NGS), NOAA. Figure 2 shows an example of the LiDAR points projected onto the corresponding orthoimage.

Figure 2. LiDAR points projected onto the orthoimage. (Images and LiDAR data provided by NGS, NOAA)

Mean Shift Segmentation

The Mean Shift Algorithm (Fukunaga and Hostetler, 1975; Cheng, 1995) is a non-parametric classification algorithm. It is usually considered as a segmentation algorithm rather than a classification algorithm because there is no direct link with one class to a segment. The procedure for Mean Shift Algorithm is given below.

1. Create a kernel with a predefined radius;
2. Calculate the center of mass for the points in this kernel; and
3. Move the kernel center to this center of mass, and repeat steps 1 to 2 until the center of mass converges.
4. The converging coordinate of the kernel center defines a segment. Each point within the radius of the trajectories that converges to the same mode belongs to this segment.
5. Repeat steps 1 to 4 until every point in the dataset belongs to a segment.

This Mean Shift Algorithm has been used in our method for data filtering and segmentation. The elevation of the LiDAR data and the color information from the orthoimages are filtered using the mean shift filtering procedure. Afterwards, the data is imported into the Mean Shift Segmentation Algorithm. The details are described in the following.

When we use the Mean Shift Algorithm for data filtering, the above-mentioned procedure has been employed. An additional step in this procedure for the consideration of minimizing the number of segments is to assign a new value (elevation or color, for example) to every point in a particular segmented group. For example, for all the LiDAR points segmented as a group, the same elevation value is assigned to each point in the group. The similar procedure is used when we filtering the color information. We assign every point in the group the same color as the mode. In this procedure, one parameter needs to be defined for each elevation filtering and color filtering. These parameters are the radius of the kernel.

After data filtering, the Mean Shift Algorithm is employed again for segmentation. The input for the segmentation is the seven dimensional feature vectors [*X, Y, Z, R, G, B, PD*] corresponding to the LiDAR points. *X* and *Y* are the horizontal coordinates from the LiDAR point cloud. *Z* is the height of the LiDAR point and is used as a classification feature directly. Strip adjustment needs to be done prior to the classification if the region is contributed to by several LiDAR strips. *R, G* and *B* are the color information gathered from the registration of LiDAR data and orthoimages. *PD* is the point density. The LiDAR point density will be different on land and on water surface, as we mentioned before, so we consider it as a feature in our segmentation method. This feature is especially useful in tidal flat regions. In this research, only the last return of each laser beam has been used; therefore, multiple returns of the LiDAR point will not create point density variance.

Using this seven-dimensional analysis, the Mean Shift Algorithm initially divides the point cloud into multiple groupings of water or land areas. These areas must be further refined into complete water bodies or land areas. In the previous learning process, we manually digitized a partial shoreline segment as ground truth, which is used as prior knowledge to determine which groups belong to water and which belong to land. For example, if we know that there are three groups of water points within the region where we have ground truth, then we can assign these three groups as a water surface class. These three groups are not only distributed over the training area; but also appear throughout the entire test region. As a result, we can find all the points belonging to the water surface in this entire test region.

Boundary Tracing

Using the above-mentioned Mean Shift Algorithm, all the LiDAR points have been segmented and classified as points either belonging to water or to land. The next step is to trace the boundary of the segmented LiDAR point groups to obtain a shoreline. Because there might be no LiDAR points on the water surface due to the scattered and weak reflection of the water surface, it will create lots of error boundaries if we trace the boundary of the LiDAR points belonging to water surface. Therefore, we select all the points that are not classified as water surface, that is to say, those points belonging to the land, for tracing the boundary and finally extracting the shoreline.

The algorithm used to trace the boundary is a modified convex hull algorithm (Sampath et al., 2007) based on the convex hull algorithm proposed by Jarvis (1977). Due to the complexity of the LiDAR point distribution within the coastal area, one constraint has been added to the algorithm, that the angle between two edges must be large than 60 degrees.

EXPERIMENTAL ANALYSIS

To evaluate the performance of the developed method, experimental analysis was conducted using the LiDAR data and the orthoimages collected simultaneously in Portsmouth, New Hampshire. Details of the experimental process are described below.

Experimental Data

The data used in our experiment is form Portsmouth, New Hampshire. It was provided by the NGS, NOAA and includes the LiDAR data and corresponding orthoimages collected simultaneously on June 8, 2008 around noon UTC time. The orthoimages were acquired while the tide was in MLLW (Mean Lower Low Water). For the LiDAR data, the horizontal accuracy is about 2 m, and the vertical accuracy is about 0.3 m. Nominal point spacing is about 1~2 m. The resolution of the orthoimage is 0.5 meter.

The test region is an isolated island called Seavey's Island located near the mouth of the Piscataqua River (Figure 3a). The reason for using this test region was because of its combination of man-made docks, shipyards, natural sand beaches, and rocky shores. Another challenge is the complexity of the near-shore terrain. Figure 3b shows an example of a local region where shoreline extraction is difficult even for human eyes.

a) b)

Figure 3. Images of the test region a) Seavey's Island in the Piscataqua River. b) Example of the difficulty of extracting shorelines. (Image provided by NGS, NOAA)

Shoreline Comparison

There are three local regions that have been studied in our experimental analysis. The first region is a man-made dock without any vessels inside the dock (Figure 4a). The length of the shoreline in this region is about 352 m. The second region is a man-made breakwater (mole) along the shore (Figure 4b). The shoreline length of this region is about 844 m. The third region is a small island with a combination of natural rocky and sandy shoreline (Figure 4c). The shoreline length is about 1.01 km. For the purpose of shoreline comparison, we manually digitized the shoreline (white lines) on the orthoimages; these manually digitized shorelines will be used as ground truth for comparison purposes.

To compare the shoreline extracted from the Mean Shift Algorithm with the ground truth, the following procedures were employed.

1) Select the manually digitized shoreline as reference line;
2) Set an interval for calculating the error distance (one-meter intervals have been used in our experiment);
3) Create nodes on the reference line using the interval defined in step 2;
4) Draw transect lines on every node perpendicular to the reference line;
5) Search for the intersection point of the transect line and the second shoreline; and
6) Calculate the distance between the intersection point and the node on the reference line.

This procedure is illustrated in Figure 5.

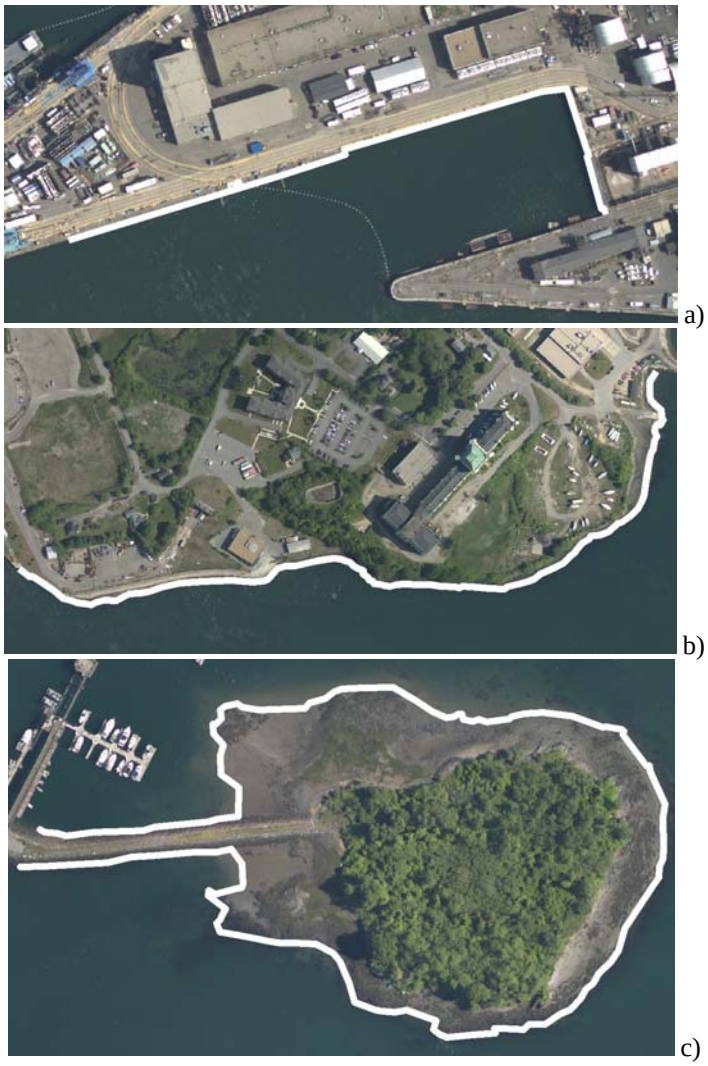

Figure 4. Examples of test regions overlaid with manually digitized shorelines (white lines): a) test region 1; b) test region 2; and c) test region 3. (Images provided by NGS, NOAA)

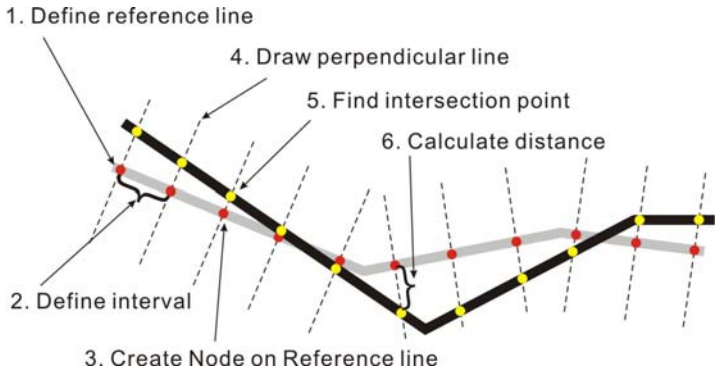

Figure 5. Shoreline comparison.

Analysis of Results

The detailed comparison results are listed in Table 1 and illustrated in Figure 6. In Figure 6, the white lines represent the manually digitized shoreline and the pink lines represent the shorelines extracted using the Mean Shift Algorithm.

In the first region, the shoreline is mainly created by the man-made dock. There is a clear separation between the water body and the land in both the orthoimage and the LiDAR data. Consequently, the extracted shorelines do not generate a large error (0.56 m RMSE) within that region. The maximum error is created by the equipment and maintenance docks (Figure 6a).

In the second region, the shoreline was created by the man-made breakwater. In this region, human eyes can distinguish the shoreline easily, and the result also shows that the algorithm obtains good results (0.68 m RMSE) in this type of coastal environment. The largest error happens in the location shown as red arrow in Figure 6b, where the breakwater material is different from that of the other breakwater and shows a different color in the orthoimage. The color difference of the breakwater causes the land/water classification to fail.

The third region is a guitar-shaped natural sand and rock island. The challenge for this region is to determine the shoreline in the natural sand beach and tidal flat area. The water/land intersection can not be easily identified even by human eyes. The RMSE is the worst in all these three regions. But the accuracy (1.57 m RMSE) is still within the nominal point spacing (1~2 m) of the LiDAR data point cloud. The largest error (red arrow in Figure 6c), about 11 m, occurred in the tidal flat area (Figure 6c). The reason for this huge error is that the elevation in tidal flat area is almost the same and the color of the rock is similar to that of the water.

Table 1. Shoreline comparison results for the three test regions

Region	*Length*	*RMSE*	*Maximum Error*
Region 1	352 m	0.56 m	3.86 m
Region 2	844 m	0.68 m	4.59 m
Region 3	1,010 m	1.57 m	11.52 m

Figure 6. Shoreline comparison: the shorelines with the largest error in: a) region 1, b) the region 2, and c) region 3.

DISCUSSION AND CONCLUSIONS

This paper investigates shoreline extraction from the integration of LiDAR data and orthoimages using a Mean Shift Algorithm. The experimental results show that, with the employment of orthoimages, this method can extract shorelines having good accuracy from LiDAR data. The accuracy of the extracted shoreline is better than the nominal point spacing of LiDAR data in general. In an area rich in man-made objects, the accuracy is around 0.5 m, while in areas with nature shoreline, the accuracy is around 1.5 m.

The timing of data acquisition is critical to shoreline extraction due to changes in the water surface caused by tidal action. In the future, we will evaluate the vertical accuracy of the extracted shorelines by comparing them to actual water-level observations from the closest gauge stations. In addition, the determination of parameters in the mean shift filtering and segmentation procedure has been done manually. We will study how to determine these parameters automatically.

ACKNOWLEDGEMENTS

This research was support by the Ohio Sea Grant Program (R/CE-010). We appreciate the collaboration of NGS, NOAA for providing the New Hampshire LiDAR data set for our experiment.

REFERENCES

Cheng, Y., 1995. Mean shift, mode seeking, and clustering, *IEEE Transactions Pattern Analysis and Machine Intelligence*, *17*(8):790-799.

Comaniciu, D., and P. Meer, 2002. Mean shift: A robust approach toward feature space analysis, *IEEE Transactions Pattern Analysis Machine Intelligence*, *24*(5):603-619.

Fukunaga, K., and L.D. Hostetler, 1975. The estimation of the gradient of a density function, with applications in pattern recognition, *IEEE Transactions on Information Theory*, *IT-21*(1):32-40.

Ingham, A.E., and V.J. Abbot, 1974. *Hydrography for the Surveyor and Engineer*, BSP Professional Books, London.

Jarvis, R.A., 1977. Computing the shape hull of points in the plane, *Proceedings of the IEEE Computing Society Conference on Pattern Recognition and Image Processing*, New York, 1977, pp. 231–241.

Li, R., R. Ma, and K. Di, 2002a. Digital tide-coordinated shoreline. *Journal of Marine Geodesy*, *25*(1):27-36.

Li, R., K. Di, and R. Ma, 2002b. 3-D shoreline extraction from IKONOS satellite imagery. *The 4th Special Issue on Marine & Coastal GIS, Journal of Marine Geodesy*, *26*(1/2):107-115.

Liu, H., D. Sherman, and S. Gu, 2007. Automated extraction of shorelines from airborne light detection and ranging data and accuracy assessment based on Monte Carlo simulation, *Journal of Coastal Research*, *23*:1359-1369.

Robertson, W.V., D. Whitman, K.Q. Zhang, and S. P. Leatherman, 2004. Mapping shoreline position using airborne laser altimetry. *Journal of Coastal Research*, *20*:884-892.

Sampath, A., and J. Shan, 2007. Building boundary tracing and regularization from airborne LiDAR point clouds, *Photogrammetric Engineering and Remote Sensing*, *73*(7):805-812.

Slama, C.C., C. Theurer, and S.W. Henriksen, 1980. *Manual of Photogrammetry*, American Society of Photogrammetry, Falls Church, VA.

Stockdon, H.F., A.H. Sallenger, J.H. List, and R.A. Holman, 2002. Estimation of shoreline position and change using airborne topographic LiDAR data, *Journal of Coastal Research*, *18*:502-513.

Woolard, J.W., M. Aslaksen, J. LT Longenecker, and A. Ryerson, 2003. Shoreline mapping from airborne LiDAR in Shilshole Bay, Washington, *Proceedings of the National Oceanic and Atmospheric Administration (NOAA) National Ocean Service (NOS), U.S. Hydrographic Conference*.