A Prototype Automated Toolbox for a Largescale Waterbody Detection Algorithm using Liner Airborne LiDAR

[1]Partha P. Acharjee, [1]Venkat Devarajan, and [2]Collin McCormick
[1]The University of Texas at Arlington, Texas, [2]National Resources Conservation Service, Fort Worth.

Abstract*— Water surface mapping is an important feature extraction application of linear airborne LiDAR data. Remote sensing capabilities, distinguishing characteristics of laser returns from water surface and specific data collection procedures provide LiDAR data an edge in this application domain. Additionally, co-registered elevation and intensity information uniquely enable the technology to simultaneously sense geometrical and optical properties of the reflecting surfaces. Furthermore, water surface mapping solutions must work on extremely large datasets, from a thousand square miles, to hundreds of thousands of square miles. National and state-wide map generation/upgradation and hydro-flattening of LiDAR data for many other applications are two leading needs of water surface mapping. These call for as much automation as possible. Researchers have developed many semi-automated algorithms using multiple semi-automated tools and human interventions. While there is justification for skepticism over claims of total automation, the research community is far from exhausting the potential for much more automation in this field. This reported work describes a consolidated algorithm and toolbox developed for large scale, automated water surface mapping. Geometric features such as flatness of water surface, higher elevation change in water-land interface and, optical properties such as dropouts caused by specular reflection, bimodal intensity distributions were some of the linear Lidar features exploited for water surface mapping. Large-scale data handling capabilities are incorporated by automated and intelligent windowing, by resolving boundary issues and integrating all results to a single output. This whole algorithm is developed as an ArcGIS toolbox using Python libraries. Testing and validation are performed on a large datasets to determine the effectiveness of the toolbox and results are presented.

***Index Terms*— LiDAR, Solar power, Shadow mapping**

I. INTRODUCTION

A water-body is defined as any physiographical feature containing water, which includes a pond, lake, reservoir, river, sea etc. Mapping water-body boundaries is a very important and demanding requirement. Traditional water-body boundary extraction approaches require manual examination, estimation and annotation from high-resolution imagery.

Recent advances in Light detection and ranging (LiDAR) technologies have introduced an economic and high resolution earth surface mapping technique. A linear or conventional LiDAR system derives the distance of target points from a laser source by emitting laser pulses and analyzing the reflected return. Thus, a LiDAR system provides highly accurate as well as dense co-registered elevation and intensity information of the target surface. However, the size of this high-density point cloud is typically enormous, which makes manual detection cumbersome and expensive. Automated or at least semi-automated detection or feature extraction algorithms are therefore desired [2].

Most of the literature covers hydro-breakline generation in coastal areas. Only a few results were reported for river and standing or still waterbodies detection. Early model-based approaches used historic shorelines and examined cross-shore LiDAR elevation profiles to detect sudden change of elevation [3,4]. Corrected LiDAR products such as DSM, DEM were used as primary information sources; in some cases, ortho-images were also incorporated.

In recent years, two new algorithms, large-scale automated standing waterbody extraction (LASWE) and multi-scale LASWE, were proposed for in-land standing water body detection, which were demonstrated on large scale standing water bodies [5]. Our own work related to detection of waterbodies over small areas, the crux of the method is described in [6].

In this paper, our previous work on waterbody extraction was extended for large-scale application. This application was developed, keeping in mind most practical scenarios and experiences of the US Geological Survey (USGS). Manual involvement was reduced to a bare minimum; constraints on hardware and software were considered, and verities of data storage system and dataset configuration were also considered. In the following sections, the prototype automated toolbox will be described in detail.

II. PROPOSED ALGORITHM

In practical cases, LiDAR data are stored as LAS files in a directory, and area covered by a single LAS file varies in different projects. Therefore, LAS file boundaries, neighborhood information, covered area and file size were not readily available. Batch processing is the desired approach to handle these enormous datasets in a timely fashion. Therefore, the whole process flow consists three parts; a) creation of batch settings, b) water body detection, and c) post-processing results. In a large-scale application, batches were created from a list of LAS files, then each batch was treated separately, and output results were post-processed and merged to the final output. Details of the process are shown in Fig.1.

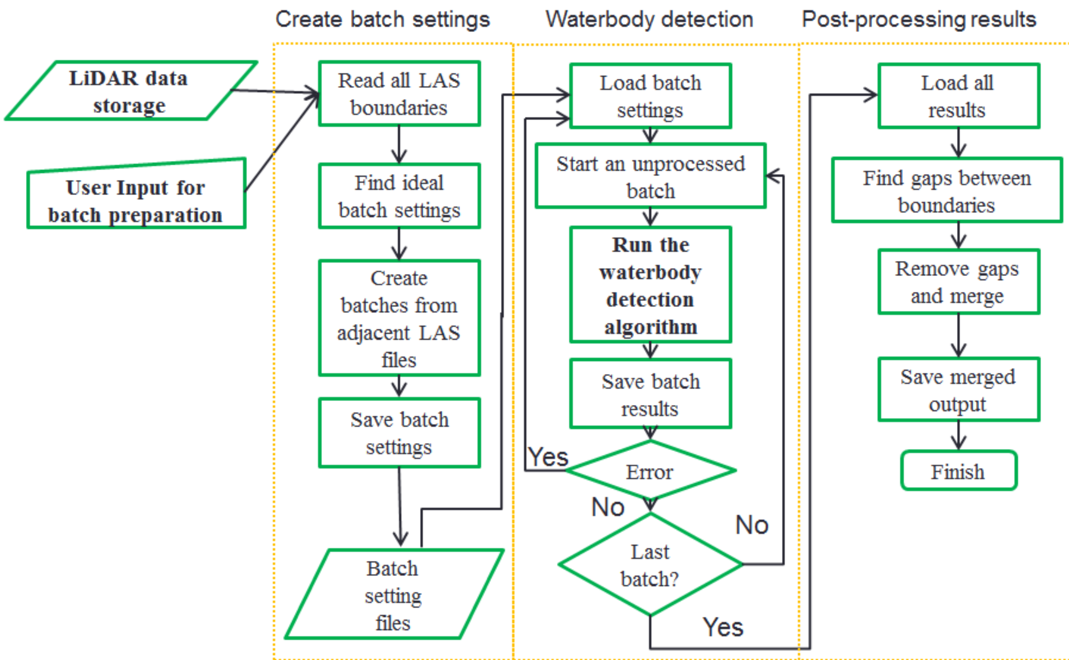

Fig.1: The overall flowchart of the proposed large-scale waterbody detection toolbox.

A. Create Batch Settings

After receiving all the user inputs, boundaries of all the needed LAS files were are to create a project. Based on the geo-location, LAS files are numbered as rows and columns where (m, n) pair is the position of a LAS files, and M and N is the total number of rows and columns of LAS files in that project. The graphical interface for user input is shown in Fig.2. The location of the LAS files, output directory, dataset name, the unit of elevation, batch types, batch dimension, and few other parameters were taken as inputs. The user can also set the highest number of LAS files in each batch too. The user can also select the options to create a single batch using all LAS files, created batches from file name, and to create a batch for each LAS files. Larger batch size ensures more available information in each batch and better water body detection. However, limitations of the host computer system capabilities determine the batch size.

Based on the user input and project structure, LAS files are grouped in batches, and a list of LAS files in each batch is saved. Additionally, all user settings are also saved in the same file called batch-setting file. For a very large project, the computing system may crash, hang or be turned off due to some circumstances. In those cases, the batch setting file can be reloaded to resume the run without giving the user inputs.

B. Waterbody detection

In this stage, the batch setting file is loaded in the tool, from which the list of batch, LAS files in each batch and the status of the batch are available. This information is fed to the water body detection algorithm. The detailed flowchart of the method is shown in Fig.3, and detailed description of the algorithm is available in [6]. In order to understand the problems associated with processing very large files, a short summary of the method from [6] is given below.

From the batch setting, the listed LAS files are used to create three rasters. The drop-out raster had the voids in the LAS file, where no LiDAR return is available. The elevation raster provides the physical information including x, y and z, and intensity raster has the LiDAR return intensity information. Based on previous observations of large water bodies and their LiDAR returns, it was noted that a) every water body has at least a small amount of drop-outs, b) waterbody elevations don’t vary much, c) elevation difference of water-land interface is higher, and d) water body intensity has a bi-modal distribution and most likely has a low intensity.

Therefore, seeds (starting points) are generated from the drop-out and low intensity regions. Angular filtering step provide the measure of elevation changes in the terrain.

Now, starting from the largest seed, a small angular value is set to grow the region around the drop-out, this means only area surrounding the seed is included as water if the elevation difference between the seed and the neighboring pixels is below a certain threshold. Then, the cumulative intensity distribution of the added region is compared with the current intensity distribution of the water-body region using Kolmogorov-Simonov test to check is there is any change to this distribution pattern. This is continued until there is a dramatic change in the distribution, which indicates that some high intensity region is detected (such as the banks of the water body), This situation is equivalent to ‘virtual flooding’.

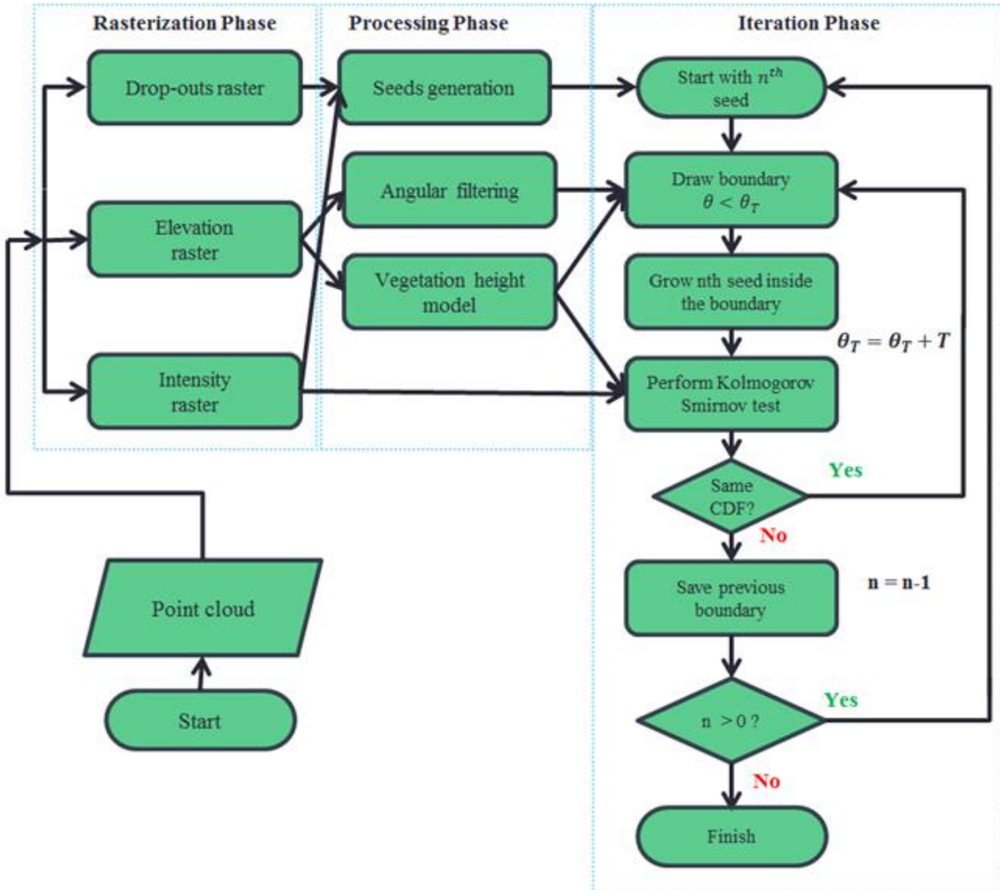

Fig.2: The details flowchart of the waterbody detection stage for a single batch.

The whole process is stopped after all seeds are processed. The result of this batch is saved, and all memory cleared to start processing the next batch. To embellish this short description of the method, interested readers are encouraged to read our previous paper in which much more details are provided [6].

The focus of this paper is very large projects. For very large projects, hundreds of batches may be generated in the whole project. The system or the machine may crash in the middle of the project— in those cases, the batch setting can be loaded in the tool to resume the work from the same batch from where it left off – saving much computation time. If all batches are processed, then the post-processing stage was begun.

C. Post Processing

In post processing, results from all batches are loaded to merge in order to obtain the final result for the whole area. In many cases, same waterbody may extend into multiple batches. In those cases, a thin gap would appear in the batch boundaries. In the post-processing step, thin gaps between two waterbodies from two different batches were detected and removed. These two separate waterbodies were then merged and considered as a single waterbody.

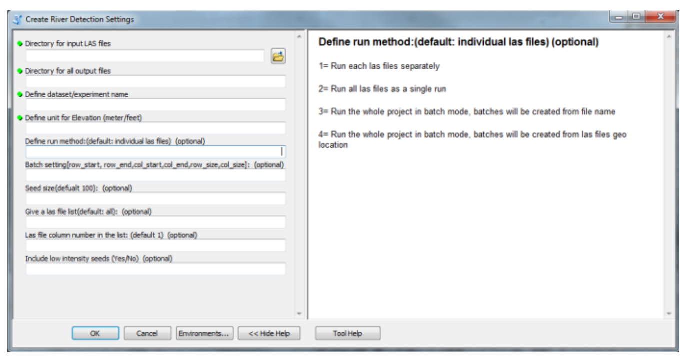

Fig.3. The graphical user interface for the batch creation inputs. This tool is developed as an ArcGIS toolbox using python.

III. Results

The proposed method was applied on a 600 km^2 area in Salisbury, Maryland. A 2m-by-2m elevation raster was created from the corresponding LiDAR point cloud, where data density was 4 points/m^2. Waterbodies detection results are shown in following figures.

In Fig.4, the overall detected waterbodies are shown in white, boundaries of LAS files are shown in red gridlines. Batches are made using a maximum of 25 LAS files—five rows and five columns. Boundaries of two batches are shown in green; maximum area of a batch is 56 km^2.

The list of batches was created in the first stage and saved as a batch setting file. In the second stage, the batch setting file was loaded into the tool, which sequentially runs each batch. Water bodies detected in each batch are saved as a separate file. In last stage, results of all batches are loaded to merge as a single output. When same waterbody extended through multiple batches those were detected as separate waterbodies by the main program and a thin separation line was appeared at the batch boundary. In this stage, those thin lines were detected and removed, and adjacent waterbodies are merged to detect a single larger water body. In Fig.5, the thin lines between waterbodies from two separate batches are shown (left), and the merged waterbody is shown in the right.

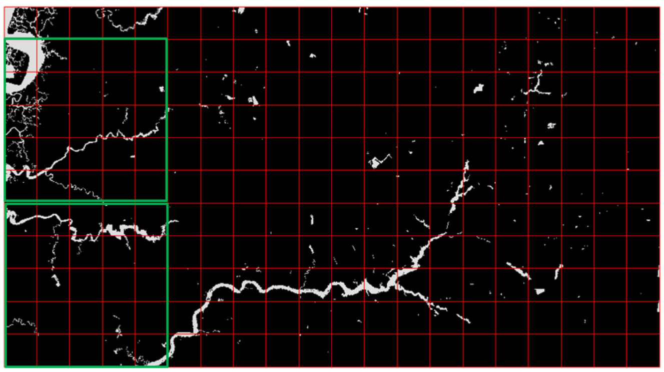

Fig. 4. Detected waterbpdies are shown in white color, LAS files boundaries are shown in red grid, and boundaries of two batches are shown in green.

Fig. 5. The thin line in batch boundary if a waterbody is extended in two batches is shown in left figure. In right figure, the merged waterbody is shown after the removal of the thin line.

In Fig.6 and Fig.7, a few enlarged views of the detected waterbodies are shown for better visualization. Detected waterbodies are shown in blue, and overlaid on areal color images. In Fig.6, we see that small channels of waterbodies were detected.

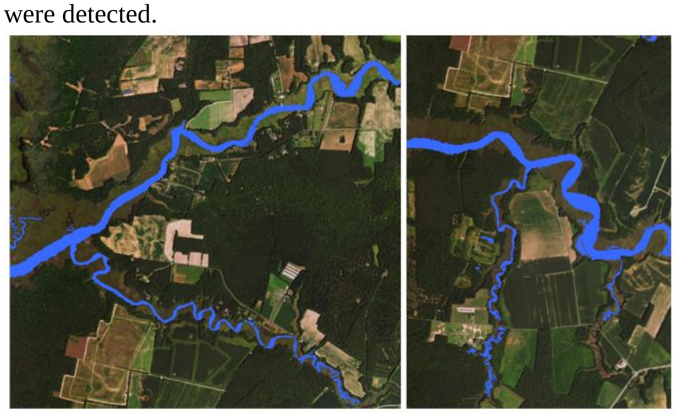

Fig. 6. Detected waterbodies are shown over areal images in blue color. Smal river branches are detected in these figures.

In fig.7, small braches from wide rivers were detected as well. Small, complicated river network was also detected. Bridges were not included inside the waterbody area.

Fig. 7. Detection result of wide river, complicated river network, and bridges are shown in this figure.

IV. Conclusions

In this paper, we proposed an algorithm for waterbody detection in large-scale application. This is an extended version of a small-scale water body detection tool. This algorithm has three stages. First one is intelligent batch processing, which automatically grouped adjacent LAS files in a single batch to deal with system constraints. In second stage, each batch was processed serially, and all results were saved. In the final stage, interim results were merged after resolving boundary issues. In future, more large-scale validation will be performed and necessary modification will be integrated to address upcoming challenges.

References

[1] A. Antonarakis, K. S. Richards, and J. Brasington, "*Object-based land cover classification using airborne LiDAR*," Remote Sensing of Environment, vol. 112, no. 6, pp. 2988–2998, 2008.

[2] A. Brzank, C. Heipke, J. Goepfert, and U. Soergel, "*Aspects of generating precise digital terrain models in the Wadden Sea from lidar–water classification and structure line extraction*," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 63, no. 5, pp. 510–528, 2008..

[3] B. Höfle, M. Vetter, N. Pfeifer, G. Mandlburger, and J. Stötter, "*Water surface mapping from airborne laser scanning using signal intensity and elevation data*," Earth Surface Processes and Landforms, vol. 34, no. 12, pp. 1635–1649, 2009.

[4] J. Smeeckaert, C. Mallet, N. David, N. Chehata, and A. Ferraz, "*Large-scale classification of water areas using airborne topographic lidar data*," Remote Sensing of Environment, vol. 138, pp. 134–148, 2013.

[5] G. Toscano, and V. Devarajan, "*A LiDAR-based auto hydro break-line generation algorithm for standing water bodies*," PhD dissertation, Uni. of Texas Arlington, 2015

[6] P. Acharjee, G. Toscano, C. McCormick, and V. Devarajan, "*Performance analysis of a novel algorithm for large-scale water-body surface mapping using elevation and intensity of LiDAR data.*" ASPRS conference, Fort Worth, Texas, 2016.