LIDAR BARE-EARTH MODELING OF OVERHANGING CLIFFS – EXTENDING 2.5-D LIDAR CLASSIFIERS TO HANDLE 3D SURFACE CLASSIFICATION PROBLEMS

Robert T. Pack, Keith Blonquist, Brad Carter
Department of Civil and Environmental Engineering, Utah State University
Logan, Utah 84322-4110

ABSTRACT

Lidar point filters that separate vegetation points from bare-earth terrain surfaces typically operate along vectors aligned with either the lidar shot direction or with the vertical. For example, when using a "last return (echo)" filter, a bare-earth point can be separated from its neighbors by evaluating multiple returns positioned along the shot vector. Alternatively, a vegetation height filter is evaluated along a vertical vector. Also a bare-earth filter typically operates on a 2.5-D surface that is in keeping with the usual goal of creating a 2.5-D digital elevation model. Thus, these constraints exclude the applicability of these filters to 3-dimensional terrain models that include vertical and overhanging cliff faces. At a local scale, the steep face of a cliff can fool a point filter into thinking it is the steep face of a tree. Moreover, an overhanging face presents a ceiling and roof surface that can cause a standard filter to fail entirely. In this paper we address the problem of adapting the host of existing 2.5-D filters to operate along a vector more normal to a steep terrain surface, or even an overhanging one. We thereby extend the capability of standard filters to the classification of truly 3D surfaces. This work is motivated by our experience in lidar mapping the canyon country of Utah and Arizona where vertical cliff faces and overhangs are common.

Our new approach follows the following steps: (1) a preliminary bare-earth classification is completed using a standard filtering algorithm; (2) the resulting TIN is then analyzed to detect if triangles with normals pointing near horizontal are present, i.e. the triangles are near vertical; (3) if so, the point cloud is rotated in 8 cardinal directions yielding a new point cloud for each rotation; (4) the rotated point clouds are then classified using standard filtering algorithms; (5) these classifications are then combined and applied to the original un-rotated points. The presumption of step (4) is that one of the eight rotations causes the normal vectors for a given vertical or overhanging cliff-face to be upward pointing and therefore capable of being classified successfully with a 2.5-D filter. Step (5) optimally combines classes in order to minimize Type I and Type II errors. The resulting marked improvement in classification results is shown and discussed.

Keywords: lidar, classification, 3-D, cliffs, steep, transformation

INTRODUCTION

Classification of lidar point clouds to separate bare-earth from vegetation is a common application in lidar mapping and many lidar point filters have been developed for this purpose. These filters typically operate along a vector aligned with the lidar shot direction or with respect to a vertical vector. A "last return (echo)" filter is an example of operation along the shot direction, as a bare-earth point can be separated from its neighbors by evaluating multiple returns positioned along the shot vector. Alternatively, a vegetation height filter is typically evaluated along a vertical vector.

Many existing bare-earth filters operate on a 2.5-D surface, working toward creating 2.5-D products, such as digital elevation models and orthophotos. Our experience has found that a filter designed with 2.5-D assumptions can give poor results when applied to 3-dimensional terrain models that include very steep, vertical, or even overhanging cliff faces. This limited performance is due to several reasons. First, at a local scale, the steep face of a cliff can be hard to distinguish from the steep face of a tree. Moreover, an overhanging face presents a ceiling and roof surface that can cause a standard filter to fail entirely.

The routine for classifying ground points in TerraScan (Soininen, 2009) is an example of a 2.5-D filter. The routine works by iteratively building and refining a triangulated surface model, or TIN. Initialization is done by classifying some local low points as ground points, where the size of the local area is controlled by a "Maximum Building Size" parameter. Given these initial points, a TIN is created, and it is assumed that the triangles in the initial TIN are mostly below the ground since their vertices are the lowest points in a localized area. The routine then starts refining the model by iteratively adding additional points which lead to additional triangles. Through several iterations the model begins to follow the true ground surface more closely. The routine is based on several iteration parameters which determine how close a point must be to a triangle in the TIN before adding that point to the model. An "Iteration angle" parameter is defined as the maximum allowable angle between a point, its projection on the triangle plane, and closest triangle vertex. A small iteration angle prevents the filter from adding points to the model unless they are rather close to a triangle in the TIN, while a larger angle allows points to be added to the model when they are farther from the surface of the nearest triangle. Setting this parameter is a way of controlling how much roughness is allowed to be included in the TIN model result. This depends on the nature of the ground the vegetation associated with a project. The second parameter is the "Iteration distance" parameter defined as the maximum allowable distance between a point and the nearest triangle. This parameter helps to prevent the iteration from making big jumps upwards when triangles are large. As with the iteration angle, a larger iteration distance makes the filter more aggressive by allowing points which are further from the model to be added. The final parameter is the "Terrain angle" parameter which is the maximum allowable terrain angle in the model.

Our experience has found that the classification method described above can often fail in steep terrain, especially cliff edges and overhangs, due to its inability to iteratively add more points to the initial ground model. Figure 1 illustrates this failure on a simulated cross section of terrain. The figure shows how the triangles of the initial ground model (shown in blue) roughly approximate the surface of the terrain (shown in black). Given the 2.5-D assumption, the filter begins with points distributed evenly in the X-Y plane. As a result, the initial ground model will have large triangles with steep normals on a steep face, as compared to those on flatter terrain. The initial ground model will also tend to deviate more from the true ground surface in these areas. The subsequent iteration steps then have difficulty adding additional points to model in these areas since the iteration angle and iteration distance to additional points on the steep slope are too large. These problems can be partially overcome by increasing the terrain angle, iteration angle and iteration distance. However this leads to the inclusion of vegetation and other items as part of the ground model. In practice, choosing these parameters for a given data set can be somewhat difficult, since a passive filter will not classify ground very well in steep areas (Type II errors), but an overly aggressive filter will begin to add erroneous ground points (Type I errors).

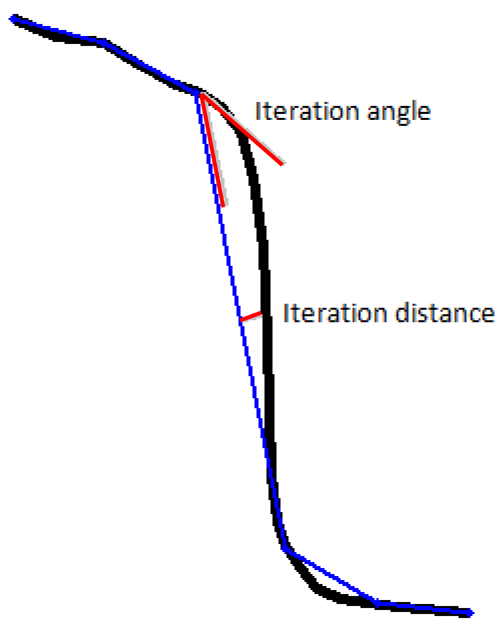

Figure 1. An initial triangulation (blue) of a cross section of terrain (black).

The iteration angle and iteration distance (shown in red) are much larger on a steep face when compared with flatter terrain.

In this paper we present a method for adapting the above filter to better classify steep areas. This work is motivated by our experience in lidar mapping the canyon country of Utah and Arizona where vertical cliff faces and overhangs are common. The paper proceeds by outlining the method in detail, then results from an actual dataset with steep terrain are shown.

METHOD

Our approach is to apply the standard filters to a set of points that are rotated in such a way as to make the steep areas of the terrain flatter. This rotation effectively allows the filter to operate along a vector more normal to the terrain surface. A 5-step approach is proposed: (1) a preliminary bare-earth classification, (2) analysis to determine the terrain steepness, (3) rotation of the data set in eight different rotation directions, (4) classification of the set of eight rotated point clouds, and (5) combination of the 8 different classification results into an optimum single classification. Each of these is steps are discussed in detail.

Step 1. A preliminary bare-earth classification is completed using a standard filtering algorithm. This preliminary classification need not be refined since it will only be used to determine if the given data set has a sufficient amount of steep terrain to warrant classification using the proposed method.

Step 2. The preliminary classification is used to create a TIN which is then analyzed to detect steep terrain. If there are a large number of triangles in the TIN with normals pointing near horizontal, this indicates the presence of triangles which are near vertical. This step is necessary when processing a large project which has been broken up into smaller data files, i.e. blocks, to avoid unnecessary processing on blocks which do not have overly steep terrain. A threshold can be set on the percentage of triangles beyond a certain degree of steepness. For example, all blocks with more than 1% of their triangles steeper than 60° could be set aside for processing with the proposed method.

Step 3. If a block is determined to have steep terrain, the points are rotated so that the direction of classification (normally vertical) is more normal to the terrain. Figure 2 shows a 2D cross section of some steep terrain and the vectors indicating the direction of a vector along which a standard 2.5-D classification filter works. Figure 3 shows the same terrain rotated 30° so that the classification vectors are more normal to the surface.

From Figures 2 and 3 the following can be noted:

1) Following the rotation of the terrain, the resulting vectors in the areas that are steep are more normal to the terrain than they were prior to the rotation.
2) The vertical displacement between the positions on the terrain marked by the vectors is greatly reduced in the steep area.
3) The projected length of the steep slope is increased following the rotation so that more vectors are placed along the slope.

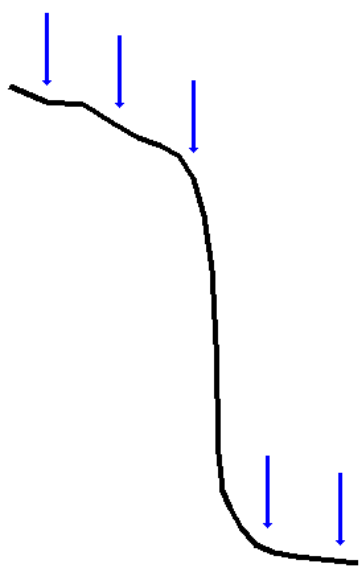

Figure 2. Steep terrain and the default direction of a 2.5-D classification filter.

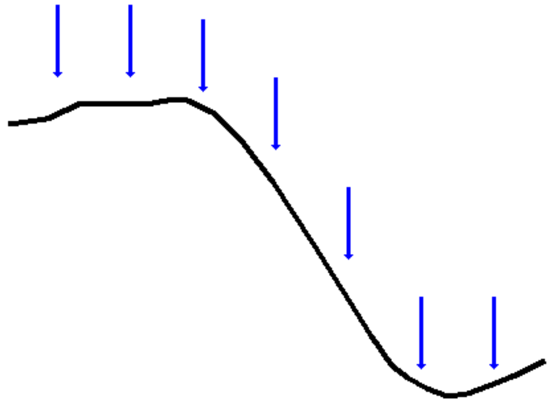

Figure 3. Rotated terrain and the resulting direction of a 2.5-D classification filter.

These differences allow an initial triangulation and resulting bare-earth classification which follows the ground more closely. For example, a 30° rotation will transform a slope of 90° down to a slope of 60°, which from experimentation has been shown to be flat enough to be classified quite well with an existing filter.

Rotations of 3D point clouds are performed about the centroid of the point cloud; a representation of this is given in Figure 4.

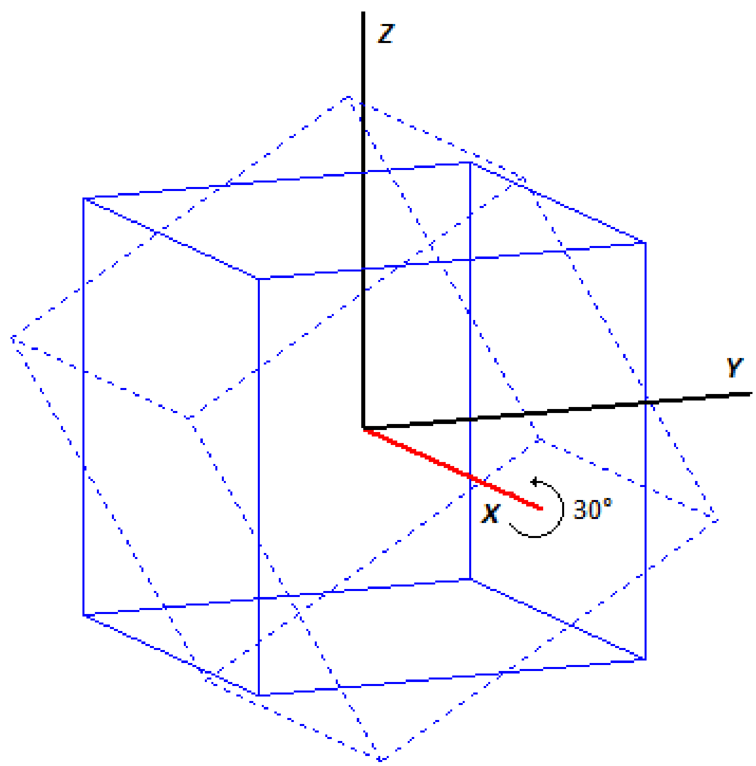

Figure 4. The rotation of a box by 30° about the X-axis.

The rotation shown in Figure 4 would be advantageous for a slope which faces the positive Y-direction since it would be "flattened" by this rotation. However, a slope facing the negative Y-direction or in the X-direction would not benefit from this rotation. In-fact a slope facing the negative Y-direction could possibly become an overhang. The point cloud is therefore rotated in eight cardinal directions to accommodate steep terrain with a multitude of aspects. The eight directions of rotation are spaced at even 45° intervals (see Figure 5), yielding eight point clouds.

Step 4. The fourth step is to classify each of the eight rotated point clouds using standard filtering algorithms. Note that the un-rotated (original) point cloud is also classified since it is best suited to areas that are flat enough. Because the point cloud has been rotated in several directions, the points on any steep face will be "flattened" in at least one of the rotated point clouds. For this reason, a conservative classification, i.e. with a low iteration distance and angle, can be used to classify each of the rotated point clouds. This helps avoid Type I errors that result from an overly aggressive filter that would have the potential of misclassifying non-ground points such as vegetation. Note that the rotated point clouds do not need to be kept in computer storage. Once they have been classified, the class for each point can be recorded separately and therefore the rotated point cloud can be immediately discarded. The geometry of the rotated points is only useful to provide a classification result.

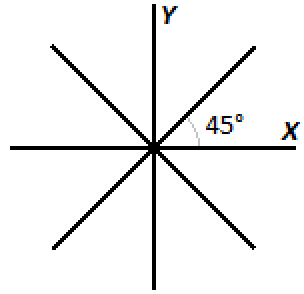

Figure 5. The 8 directions of rotation.

Step 5. The fifth and final step is to combine the classification results of the un-rotated point cloud with the results of the eight rotated point clouds. Since a conservative filter is used to avoid Type I errors, a simple Boolean OR operation can be used to combine the results. If a point was classified as ground in any of the individual point clouds, it will be classified as ground in the final result. The inclusion of all ground points in the final result helps to avoid Type II errors, making the overall filter result fairly aggressive. In other words, it is the sum of the nine conservative ground classifications.

EXPERIMENTS

The performance of the proposed method is demonstrated using lidar data collected during an airborne survey within Dinosaur National Monument near Vernal Utah. The data was collected with a Riegl Q560 laser scanner from an average altitude above ground level of about 900 meters resulting in a shot spacing of about 1 shot per square meter. The data from individual flight lines was tiled into 800 meter x 800 meter blocks. Figure 6 shows a shaded relief map of one of the blocks (Block 60) with contours at a 10 m interval. It shows some flatter areas near a small river in the northern part of the block, some flatter areas on the top of a plateau in the southwest part of the block, and some extremely steep terrain including cliffs and overhangs between the flatter areas. Additionally, there is a rock rib near the northwest corner of the block.

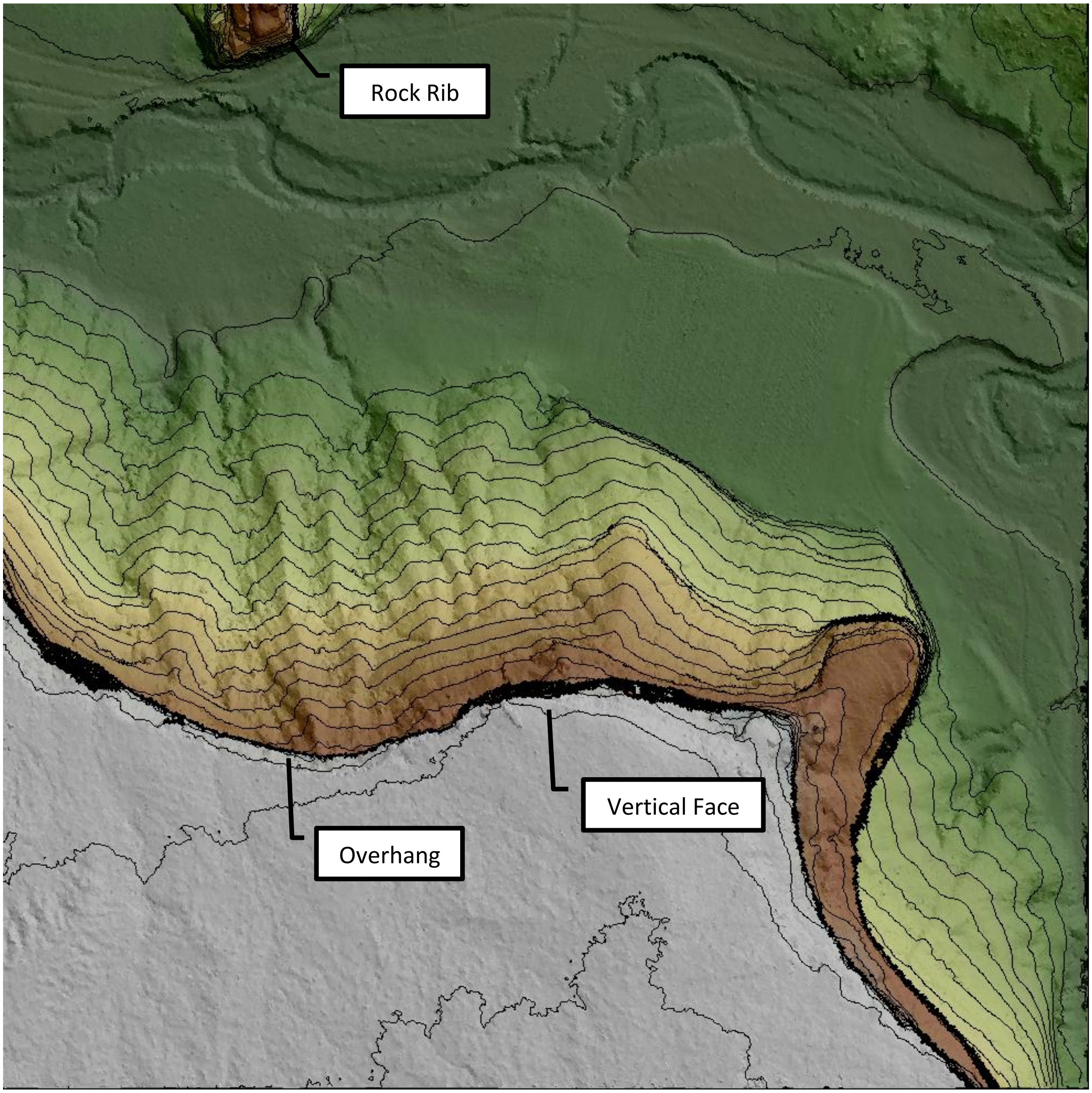

Figure 6. A contour map with shaded relief of Block 60, an 800 meter x 800 meter lidar dataset from Dinosaur National Monument near Vernal Utah.

To demonstrate the first two steps of the method, a simple ground classification was performed by selecting the lowest point in every 15 meter grid as a ground point. These ground points were then triangulated to create a TIN which very roughly approximates the ground surface. The triangles were then analyzed to see what percentage of the triangles was steeper than 60°. The centroids of the triangles are shown in Figure 7, with the steep triangles being colored red; the normals of the steep triangles are shown in blue.

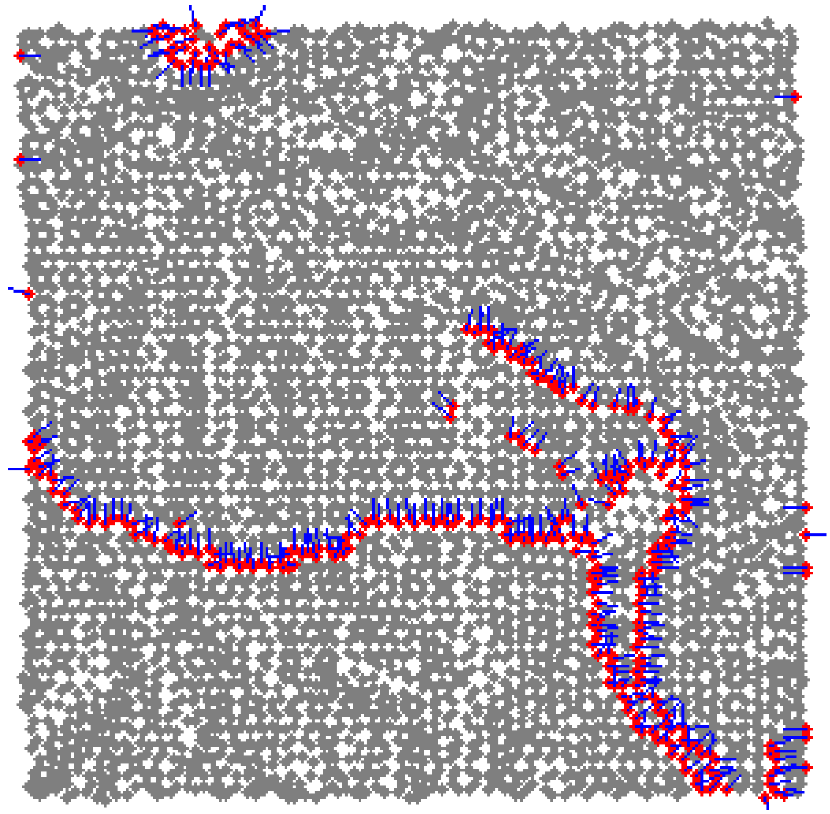

Figure 7. Triangle centroids of the coarse TIN generated in Step 1. Triangles that are steeper than 60° are colored red with their normal vectors shown in blue.

From Figure 7, it is easy to see the location of steep areas mentioned before. The percentage of steep triangles was roughly 5%, which is rather small, considering the amount of steep terrain in the block. This indicates that the threshold for identifying blocks with steep terrain should be set quite low in this type of terrain. In this case we decided that all blocks with more than 1% of the triangles should be considered. Another item in Figure 7 worth noting is the direction of the normal vectors for the steep triangles. There are normals pointing in many directions, showing the necessity of rotating the block in several directions.

The next step was the rotation of the block by 30° in each of the eight cardinal directions. Each of the nine resulting blocks (including the un-rotated block) was then classified using the TerraScan filter described above using the relatively conservative parameters given in Table 1.

In the final step the results of the separate classifications were combined using a Boolean OR operation to give the final classification results.

As a means of comparison, the un-rotated block was also classified using the more aggressive set of parameters shown in Table 1 to give an idea of the classification results obtained without using our proposed method. These parameters were chosen through the evaluation of the results of several iterations of classification. Parameters were chosen that would minimize Type I and Type II errors.

Table 1. Filter parameters for aggressive filter and passive filter.

	Aggressive Filter	Conservative Filter
Terrain angle	88°	75°
Iteration angle	20°	15°
Iteration distance	1.2 meters	1.0 meters
Max building size	10 meters	10 meters

RESULTS

Several cross sections through the point cloud have been plotted with points colored according their classification. This has allowed us to qualitatively compare the results of the proposed method with the results a "best" classification using standard methods.

Cross section 1. The first cross section that is considered is a steep area below the top of the plateau on the east side of the block. The cross section contains a steep cliff face near the top, a gradual slope below the cliff, and a steeper slope below the gradual slope. At the bottom is a flat floodplain. An overview of the cross section, and the classification results from the aggressive filter, the conservative filter, and the proposed method are shown in Figure 8.

In Figure 8, the aggressive filter (upper right) was able to classify the majority of the ground correctly except the steep cliff face. The conservative filter (lower left) was not able to correctly classify the steep face or the ground on the steep slope near the bottom of the cross section. The new method, which is a combination of the results of conservative filters from eight rotated point clouds, was able to correctly classify the steep slope and the cliff face. The poor classification results from the standard methods are quite common given the limitations of 2.5-D filter methods; in both cases the near vertical cliff face is not classified correctly.

Cross Section 2. The next cross section shows an overhang along the north rim of the plateau including a flat area at the top and a sloped area below the overhang. The location of the cross section and the classification results are shown in Figure 9. It shows that neither the aggressive nor the conservative filter is able to correctly classify the overhang, while the new method succeeds. Also, all three filters correctly classify the vegetation points at the top of the plateau and on the slope below.

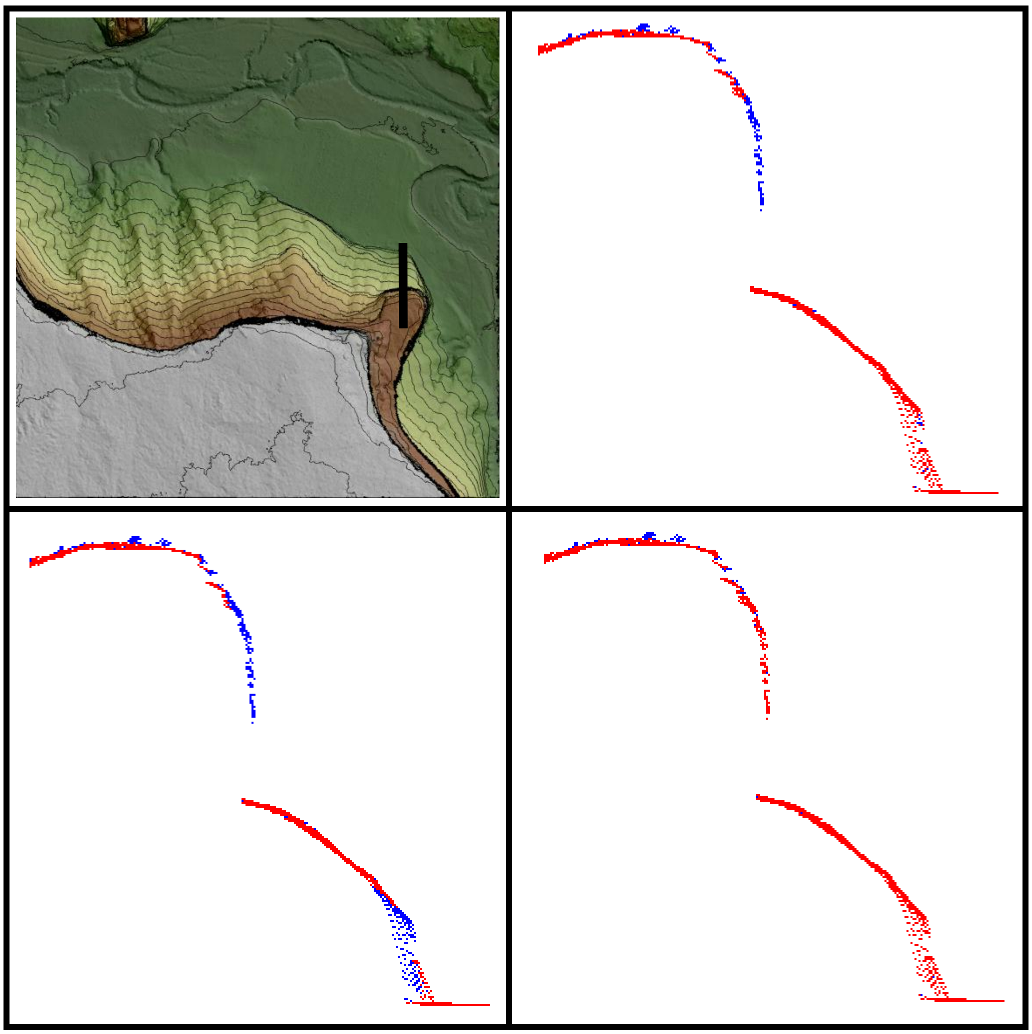

Figure 8. Cross Section 1 (small black line in upper left), the classification results from an aggressive filter (upper right), a conservative filter (lower left), and the new method (lower right). Red points are classified as ground.

This gives confidence that the proposed method is reducing the number of Type II errors without increasing the number of Type I errors.

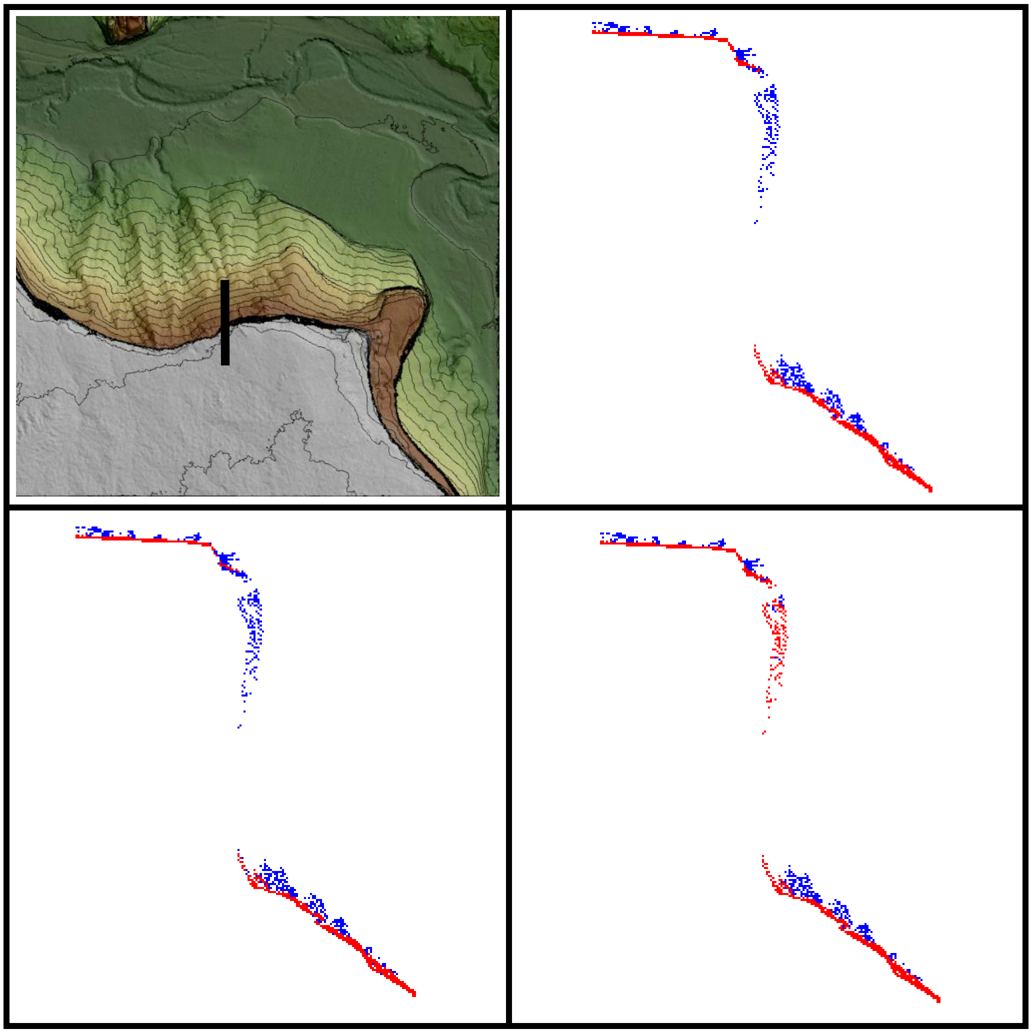

Figure 9. Cross Section 2 (upper left), the classification results from the aggressive filter (upper right), the conservative filter (lower left), and the proposed method (lower right). Red points are classified as ground.

Cross Section 3. The third cross section is of the rock rib in the northwest corner of the block. Figure 10 shows that the cross section intersects a rib with a small spire on one side. It can be seen that the aggressive classification is able to classify the majority of the ground points on the feature while the conservative classification leaves much of the feature misclassified. The new method is able to correctly classify some additional ground points that the aggressive filter misses. However, the new method is unable to correctly classify the spire feature on the west side of the cross-section. This feature shows that the method could still be improved.

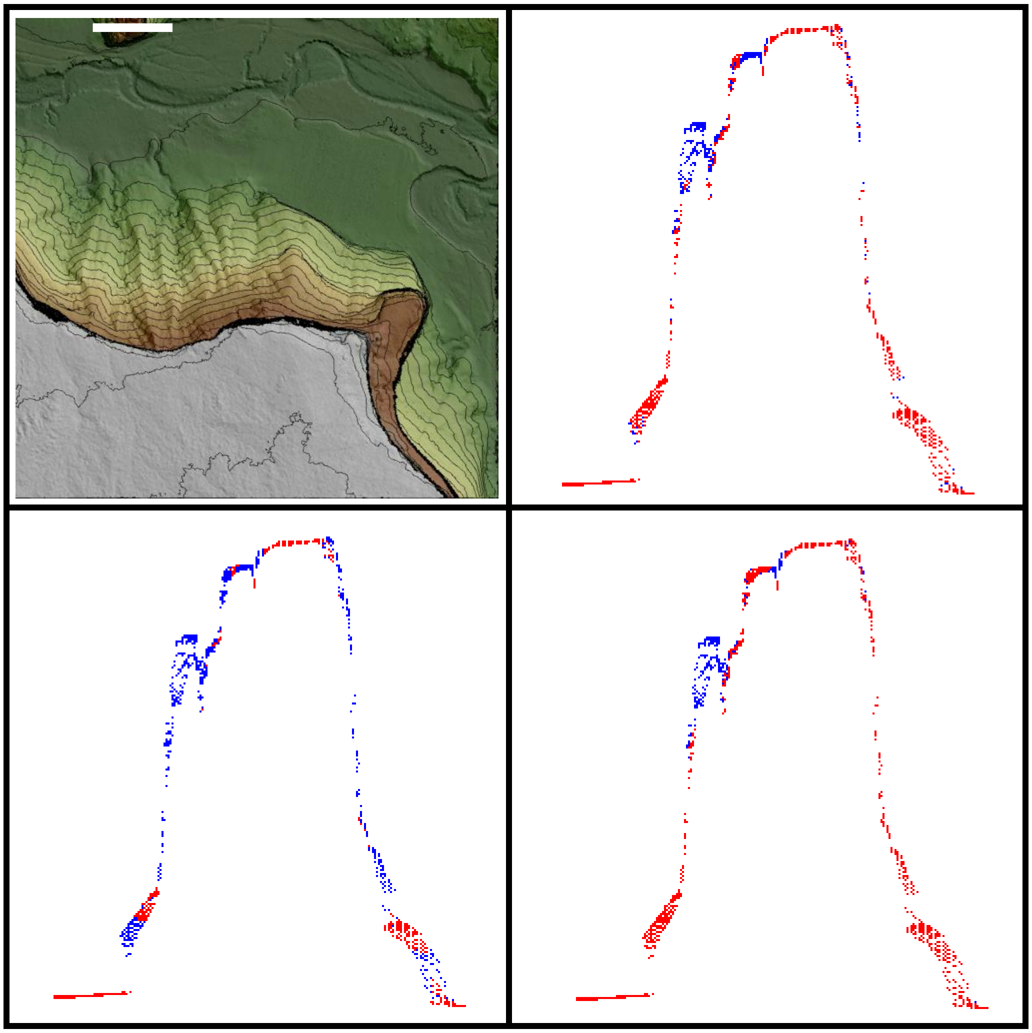

Figure 10. Cross Section 3 shown as a small white line (upper left), the classification results from the aggressive filter (upper right), the conservative filter (lower left), and the proposed method (lower right). Red points are classified as ground.

CONCLUSIONS

This paper has shown that, in all cases tested, our proposed new method markedly improves automatic bare-earth classification result in the steep terrain of Dinosaur National Monument. However, the results seen in Cross Section 3 indicate that improvements can be made. Our next step will be to apply 3-D surface Delaunay triangulation techniques. The normal vectors for each triangle even for the overhanging ones will then be used to form the basis for an equal-are warping the terrain to a flat surface. Traditional classification techniques will then be used to classify bare-earth points using the principles outlined in this paper.

REFERENCES

Soininen, Arttu, (2009). *Terra Scan User's Guide.* Terrasolid.