SOME ALGORITHMS FOR VIRTUAL DEFORESTATION (VDF) OF LIDAR TOPOGRAPHIC SURVEY DATA

R. A. Haugerud[1] and D. J. Harding[2]
[1]U.S. Geological Survey c/o University of Washington, Seattle Washington 98195
rhaugerud@usgs.gov
[2]NASA Goddard Space Flight Center, Greenbelt MD 20771
harding@core2.gsfc.nasa.gov

Commission III, Working Group 3

KEY WORDS: lidar, laser scanner, filtering, vegetation, canopy, algorithm, topography, ground surface, virtual deforestation

ABSTRACT

Lidar topographic surveys of forested terrain generate XYZ positions for laser returns from numerous points, some on the ground and some from vegetation. Extracting a ground surface model from such data requires 'virtual deforestation' (VDF), preferably by automatic means. A simple error budget for lidar topography of forested terrain suggests that the dominant source of error—and the greatest room for improvement—lies in VDF procedures.

We discuss a despike VDF algorithm that classifies returns as ground or not-ground on the basis of the geometry of the surface in the neighborhood of each return. The despike algorithm is fully automatic, effective, and can recover breaklines. It fails to identify some negative blunders, rounds some sharp corners off the landscape, and as implemented is slow. There are clear paths to improve its speed. If multiple-return data are available, a no-multiple-returns VDF algorithm robustly defines areas where all returns are ground returns. Many groups are using variations on block-minimum VDF algorithms, but these do not work well on slopes and typically require substantial human involvement to adjust block size as the fraction of ground returns changes.

Fully automatic VDF algorithms are desirable not only to minimize survey costs but also to produce topography for which all necessary interpretive biases and assumptions are explicit. The development of effective VDF algorithms has been hindered by the tendency of some commercial and academic practitioners to keep their work proprietary. Open dialogue is needed.

1 INTRODUCTION

Airborne lidar (LIght Detection And Ranging, also known as ALS—airborne laser scanning or ALSM—airborne laser swath mapping) surveys promise topographic models that are more detailed and more accurate than those obtained by traditional photogrammetric methods. The potential for improvement is especially great in heavily forested areas, where the ground is poorly illuminated and particular ground points are rarely visible on both photographs of a stereoscopic pair.

On Bainbridge Island west of Seattle, WA, an initial lidar survey (described by Harding and Berghoff, 2000) serendipitously showed evidence for recent faulting (Nelson and others, 1999) despite dense forest. Subsequently, we joined local government and other USGS researchers in the Puget Sound Lidar Consortium (*http://pugetsoundlidar.org*), a cooperative effort to contract for lidar surveys of large parts of western Washington. Though our primary focus is earthquake hazards, these data will have many uses, including mapping of other hazards, geologic mapping, transportation planning, forestry, municipal planning, and fisheries.

Puget Sound Lidar Consortium members have contracted with TerraPoint LLC for surveys using a laser altimeter that covers ±17º from nadir using a rotating pyramidal scan mirror, produces a 0.9 m diameter laser beam on the surface, and records up to 4 returns for each laser pulse with a constant-fraction discriminator pulse-detection scheme. The survey is designed to yield a uniform distribution of laser pulses across a 600 m swath with across- and along-track spacing of 1.5 m. A 50% minimum sidelap between swaths ensures that all areas are covered at least twice, leading to an average pulse density of about 1/m^2. Forest cover in the Puget Lowland includes coniferous evergreen, broadleaf deciduous, and mixed stands. All data are collected in winter months to maximize ground returns. The Consortium is purchasing all-return data, classified bare-earth returns, a bare-earth surface model, and a first-return surface model. All data are delivered in State Plane projection with English units. Surfaces are gridded to 1.8 m (6 ft) cells. Approximately 4,000 km^2 of the Lowland have been surveyed to date, with another 2,500 km^2 scheduled for survey in the winter of 2001-2002.

Initial bare-earth surface models delivered by the contractor did not appear to be the best that could be obtained from these data, which prompted us to develop a new algorithm for removing returns from the forest canopy. Because we are removing trees algorithmically, rather than with chainsaws, a colleague has dubbed this post-processing 'virtual deforestation' or VDF. TerraPoint subsequently has implemented and extended our VDF algorithm for the post-processing of Consortium lidar data. This algorithm is the subject of this paper.

An error budget for lidar topography

A topographic surface is produced by measuring, with a laser scanner, the XYZ coordinates for numerous returns, some from the ground and some not; classifying these returns as ground or not-ground (vegetation, man-made structures); and interpolating from the discrete ground returns to a continuous surface. This procedure suggests a simple error budget for the topographic surface:

$$\text{Z error} = \left[(\text{measurement error})^2 + (\text{classification error})^2 + (\text{interpolation error})^2 \right]^{1/2}$$

Measurement errors for lidar surveys are commonly estimated at about 15 cm (Z) (e.g. Huising and Gomes Pereira, 1998; Schenk and others, 1999). The average classification error is

(fraction false ground points) * (average height above ground of false ground points)

Where tree heights are many tens of meters, the classification error is potentially quite large, on the order of meters. The interpolation error is difficult to estimate without *a priori* knowledge of typical wavelengths in the topographic surface, but we do note that the probable interpretation error increases with ground-return spacing. Our experience in the Puget Lowland is that in densely forested steep areas (ravines, landslides along walls of large valleys), ground-return spacing of tens of meters commonly results in many-meter interpolation errors. Note that if interpolation error is related to surface smoothness, and if the classification of laser returns as ground or not-ground is based on smoothness of the resulting ground surface, classification and interpolation errors may be correlated and the above formula may be inadequate.

Using current technologies in forested terrain, the greatest improvement in the quality of lidar topography is likely to be achieved by reducing the misclassification of vegetation and structure returns as ground and by increasing the number of returns on the ground. The latter can be achieved both by changing the survey design (flying in leaf-off conditions, increasing pulse density) and by not misclassifying ground returns as vegetation returns.

What is ground?

In the lidar context it is useful to define the ground ("bare earth") as that surface which is continuous, is smooth, and has nothing visible below it. Note that this definition is scale dependent! With sampling at meter intervals, point returns from a hectare of forest do not constitute ground, as stem/branch/leaf returns introduce discontinuities. If the instrument could record multiple returns or the laser were aimed off-nadir, we would see many of these vegetation returns underlain by deeper returns. However, a 10 cm by 10 cm piece of tree branch, when observed with closely-spaced pulses of a millimeter-wide laser beam, may be continuous, smooth, and not underlain by any other source of returns—that is, it is "ground." Note that when observed at a large enough scale many buildings are ground by this definition, unless sharp corners catch part of a beam and lead to multiple returns.

Clearly this definition of "ground" does not meet many needs, but it (1) does match the capabilities of lidar technology and thus (2) can focus discussion of how to interpret lidar survey data. For example, this definition suggests that a filtering algorithm which preserves small details of the ground surface—pits from wind-thrown trees, glacial erratics, many break-lines—can not be expected to also identify building returns. To produce detailed, accurate bald earth surfaces (no vegetation or buildings) will require post-processing with multiple algorithms.

2 DESPIKE ALGORITHM

Smoothness—the property of the ground surface that, if sampled closely enough, it has no sharp corners—suggests a route to identifying ground points. One can search for local aberrations—points that define local strong curvatures—and remove them. Definition of ground as the lowest surface suggests that we preferentially remove points that define sharp upwards convexities. The geometry of the laser-return surface can be examined by representing it as a TIN (triangulated irregular network) constructed from the discrete returns. Because the geometry of the surface changes as we remove points, such a procedure must be iterative. That is,

```
repeat
        Build TIN
        Identify points that define strong curvatures
        Flag points as not-ground
until no or few points are flagged
```

Identifying points with strong curvatures is the nub of the problem. In ARC-INFO we have done this by

	ARC-INFO command
Convert TIN to grid1	*tinlattice*
Calculate 3x3 mean at each cell	*grid2 = focalmean(grid1)*
Convert TIN vertices to point database (Z value in item SPOT)	*Tinarc TIN cover1 point*
Calculate item SPOT2 = value of GRID2 at each point in database	*latticespot grid2 cover1 spot2*
CURVATURE = SPOT2 – SPOT	
If CURVATURE > testvalue1 or CURVATURE < testvalue2 then mark point for deletion	

`testvalue1` is chosen largely on the basis of the cell size used for grid1 and grid2. To minimize interference between neighboring points, cell size should be less than the typical point spacing for a lidar survey. With a 0.9 m laser beam diameter and average beam spacing of 1 m, we have successfully used 0.6 m (2 ft) cells. `testvalue1` is then taken at 0.2 m (0.7 ft), to accept a point on the outside shoulder of a forest road with an angle-ofrepose slope below it. Testing for `CURVATURE < test-value2` eliminates some negative blunders; this is discussed further below.

The despike algorithm can work with first returns, last returns, or multiple returns. To minimize the computational effort we use only the last return of our multi-return data. It commonly takes at least 10 iterations for the fraction of newly identified not-ground returns to drop below 0.1%, the criterion we use for convergence.

Figure 1 is a last-return surface for an area east of Seattle. Figure 2 presents the output of the despike algorithm applied to these data. Note the excellent definition of the road that traverses the scene and the good definition of the steep-sided ravine at the top of the scene. Small structures are completely removed (at **A**) or left as isolated rounded lumps (**D**, **E**). The ground surface is least satisfactorily defined in some wooded areas where few ground points remain (**B**, **C**) and at **C** this has probably resulted in truncation of the ridge crest.

Note that all the surfaces we show are produced by linear interpolation from a TIN: while other interpolation techniques may produce more realistic surfaces, conspicuous facets in a TIN-derived surface directly inform the viewer that the surveyed point density is not adequate to characterize the local curvature of the surface.

Advantages of despike algorithm

The despike algorithm creates surfaces that both look realistic and, where we have surveyed ground control, match reality. It retains large numbers of points. It requires no human intervention: to a geomorphologist concerned with understanding the processes that create the Earth's surface, this is extremely important because it means that all assumptions and biases necessary to interpret a topographic surface from raw observations are explicit in the algorithm.

If the lidar survey happens to include returns from topographic breaklines, the despike algorithm can retain them. This is an advantage over the iterative robust interpolation algorithm (Kraus and Pfeifer, 1998; Pfeifer and others, 1999), which smooths all corners.

Without complete implementations of alternate VDF algorithms (see comment below), it is not possible to judge the relative effectiveness of different algorithms at retaining ground points.

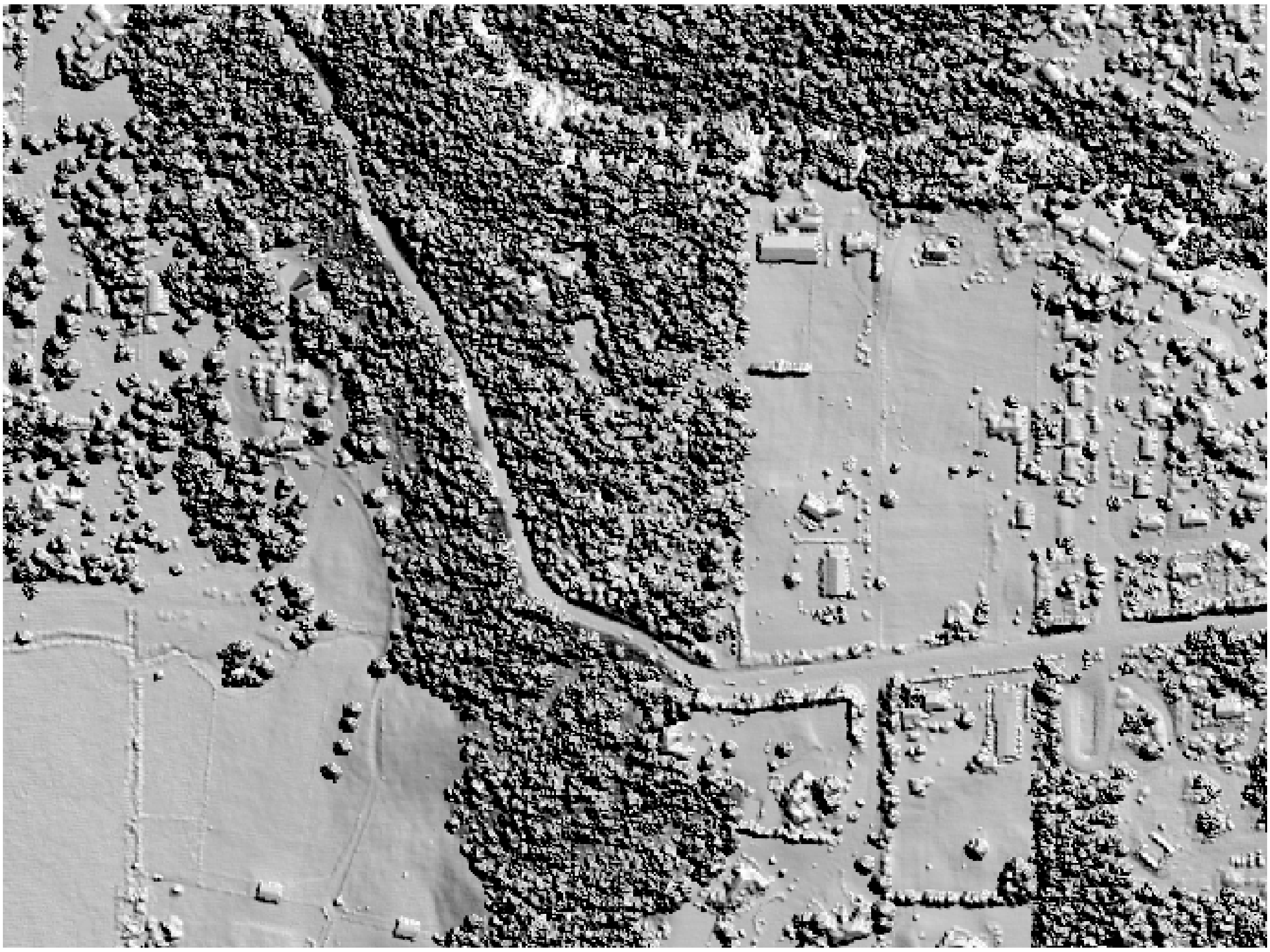

Figure 1. Last-return surface model of suburban area east of Seattle, Washington. Area shown is 850 m east-west by 760 m north-south and includes about 5 x 10^5 last-return points.

Figure 2. Surface model produced with despike algorithm, same area as figure 1. Area shown includes about 2.7 x 10^5 ground returns. See text for discussion.

The despike algorithm effectively removes small buildings and most bridges. It does not remove large-area, low-height buildings.

The despike algorithm retains more points than the block-minimum algorithms we have implemented and appears to retain more points than commercial block-minimum algorithms.

Disadvantages of despike algorithm

We have encountered three significant deficiencies of the despike algorithm.

Corner removal First, even if there are abundant returns, the despike algorithm eliminates points at the corners between near-vertical faces below gently sloping surfaces, e.g. some highway cuts in competent rock. In open (no canopy) areas this failing can be partially remedied by substituting the surface obtained with a no-multiple-returns algorithm (below). This substitution can be entirely automated.

Negative blunders The despike algorithm is especially sensitive to negative blunders. Most lidar survey data contain a few (~1 in 10^5) points dramatically lower than their surroundings and not correlated with real features.

In our experience such negative blunders are more frequent where the instrument is closer to the ground surface, are largely associated with near-nadir pulses, and are often associated with mirror-like surfaces (e.g. still water, automobiles.) Some workers (e.g. Pfeifer and others, 1999) have ascribed these negative blunders to multiple-bounce reflections—that is, reflections off more than one object in the target area. We have found negative blunders where a second reflecting surface is not evident and speculate that some negative blunders may be artifacts generated within the altimeter receiver by high-energy returns.

Processing such negative blunders with the despike algorithm can lead to distinctive ‘bomb craters’—conical pits where surrounding valid ground points have been eliminated. Where the blunder is severe enough, pit diameter reflects the number of iterations of the despike process. Solitary negative blunders can be readily caught by setting `testvalue2` to 2 * grid2 cell size. Unfortunately, this occasionally discards ground returns from small ravines beneath forest canopy. Setting `testvalue2` to 4 * grid2 cell size catches some negative blunders (but not all) and appears to keep ground returns from small ravines. We have had some success with pre-processing data to eliminate negative blunders, and further work is warranted.

Computation time The third major disadvantage of the despike algorithm is excessive computation time. Processing ~10^6 points from a 1-km^2 area takes about an hour on a dedicated single-CPU Sun Ultra 60 workstation. There is room for improvement: (1) The slowest part of the computation is building a TIN, at which ARC-INFO is not particularly efficient. (2) Within ARC-INFO the TIN data structure is not directly accessible, forcing translation of the TIN to a grid and a point set, followed by intersection of the grid and point set to evaluate curvature at each TIN node. Moving the despike algorithm to code in which the TIN data structure is directly accessible would minimize much disk I/O and reduce the amount of calculation. (3) The data could be intelligently thinned when the initial TIN is constructed. Where there are several laser returns in close proximity the lowest of these should be retained and the remainder discarded. This is not possible within ARC-INFO.

Surface roughness

Ground-surface models produced with the despike algorithm typically have widespread short-wavelength surface roughness. Enough of this roughness correlates with land cover (rough in forest, smoother in open areas) that much of the roughness probably reflects remnant vegetation. Particularly obvious are local rough areas with maximum height above the surrounding surface of 1 to 2 m. Due to pulse duration, detector bandwidth, and limitations of the ranging electronics, dual- or multi-return laser altimeters have a detection 'dead time': for a single pulse the instrument is unable to detect a return from a surface that is located closely below a surface that yields a prior return. The minimum distance between multiple returns detected by the TerraPoint system used in our project is 1.4 m, which prevents detection of returns from ground beneath some brambles and low shrubs even though the ground is illuminated by the laser.

In addition, some of the observed surface roughness reflects measurement error. This is particularly evident for very smooth surfaces where there is inconsistent navigation between overlapping flightlines. Processing data from one flightline alone reveals pavement as smooth, whereas combining two overlapping flightlines whose data are offset vertically and/or horizontally gives pavement with an orange-peel texture.

Some surface roughness is probably true ground roughness. For example, fluvial deposition may form a nearly-flat surface on a flood plain. If the flood plain is forested, toppling of trees during windstorms produces pits where the trees were rooted and hummocks where the rotting up-turned roots drop their attached soil. Human analysts commonly omit such detail as they draw contours, and we have come to assume that 'plastic' contours and the smooth surfaces they represent are more valid.

The rough area to the left of **A** in figure 2 is probably vegetation incorrectly classified as ground. But some of the high returns in this area may be from stumps that are 'ground' by the definition proposed above.

For the geologist wishing to accurately image the surface of the (forested) Earth in order to understand the processes that formed this surface, surface roughness and the uncertainty about its origin present a conundrum. Is it real and meaningful, or should it be modeled out? To make matters worse, many tools for analyzing geomorphic surfaces presume the (perhaps unreal) smoothness associated with surface models filtered through human contouring.

3 OTHER VDF ALGORITHMS

No multiple returns

The absence of multiple returns suggests that, at least locally, the laser beam has reached ground—there is nothing visible beyond this surface. Requirements that there be no multiple returns within some distance, and that contiguous areas of no multiple returns exceed some minimum dimension, quantify the "at least locally" qualification.

To implement this algorithm:

*Parse all-return data into two lists of XY locations:*First_returns *and* Other_returns

Convert these two lists to an integer grid: if there is an "other return", cell value = 1, else if a first return, cell value = 0

Expand multiple-reflection areas by `length1`. S*hrink multiple-reflection areas by* `length2`. *(*`length1` *is greater than* `length2`, `length1 - length2` *is minimum size for no-multiple-reflection areas,* `length2` *is minimum distance to a multiple-reflection area)*

Use final integer grid as mask to cut out valid part of 1st*-return surface*

The best values for `length1` and `length2` are subject to experimentation. We suspect they should depend on canopy type, laser spot spacing, laser beam diameter, detector sensitivity, and one's tolerance for misidentification of bare earth. (If we are willing to accept a slightly higher error level, we can identify more reflections as likely to be bare earth).

Figure 3 is an image of that part of a 1st return surface identified as ground by this algorithm.

This algorithm is exceedingly robust. With appropriate parameters it rarely misidentifies bare-earth reflections as canopy and within large bare-earth areas it doesn't falsely reject any points. It does require a multiple-return scanner with a laser beam that is sufficiently wide and powerful to create a significant number of multiple returns. And, obviously, it provides no information about the ground surface that is beneath canopy.

Block-minimum algorithms

The observation that ground points should be the lowest points in a neighborhood suggests a block-minimum function as a bare-

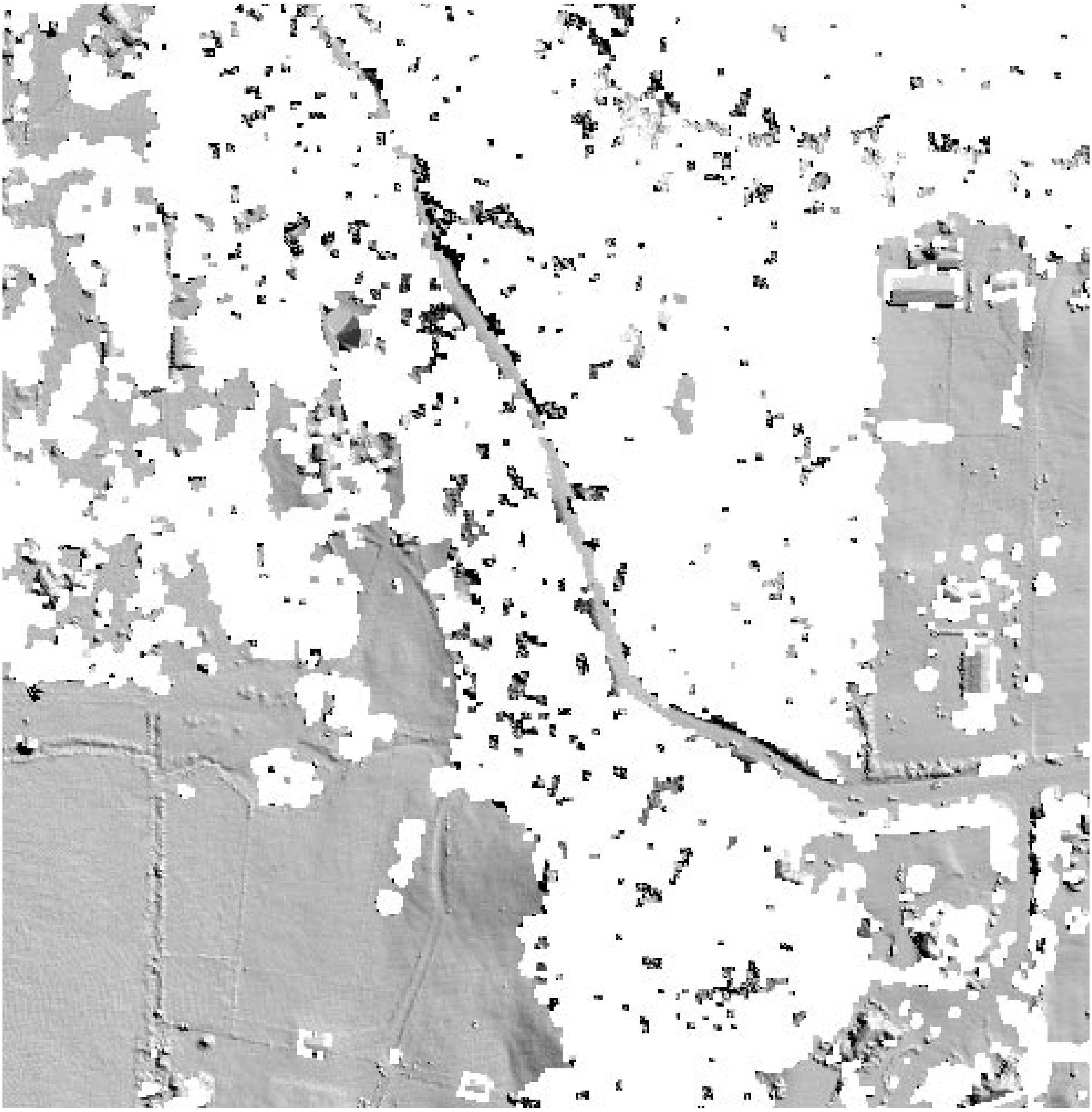

Figure 3. Surface model obtained with no-multiple-return algorithm, same area as figures 1 and 2. White areas are undefined. Model calculated with 2 ft (0.6 m) grid cells, `length1` = 10 ft (3 m) and `length2` = 6 ft (1.8 m).

earth filter. Implementations and extensions of block-minimum functions have been described by Kilian and others (1996), TerraScan (1999) and Hansen and Vögtle (1999). Proprietary algorithms used by some North American lidar survey enterprises appear to be block-minimum algorithms.

A block-minimum algorithm can be enhanced by accepting as ground returns those points that are no higher than some specified amount above the block-minimum surface. Or one can use a larger block size and take the 5th percentile (or other) elevation, thus rejecting negative blunders.

A block-minimum algorithm implemented in the raster domain can be computationally rapid, but this loses positional (XY) accuracy. The derived surface can have a tweedy appearance in steep areas. It can be biased low on slopes, as the lowest points at the edge of a block are attributed to all of a block.

Block minimum algorithms have two fundamental weaknesses. First, the necessary block size must be inversely proportional to the ground return density: areas with low density of ground returns require larger blocks. For optimum results the block size needs to be adjusted, typically with the intervention of a human operator. Second, and less obvious, block minimum algorithms contain the implicit assumption that the ground is horizontal. They can produce acceptable results in low-slope areas; we note that some of the successful users of block-minimum VDF algorithms come from the Gulf Coast of North America.

Iterative linear prediction

Kraus and Pfeifer (1998) and Pfeifer and others (1999) described a VDF algorithm that attempts to explicitly model measurement error, derive a ground-surface model, and thus classify returns as ground or not-ground. Like our despike algorithm, it uses the smoothness of the ground as a guide to building a ground-surface model. Unlike the algorithms described above, it fits a polynomial surface to weighted laser returns, increasing the smoothness of the surface and gaining the possibility of modeling (and thus removing) random measurement errors at the price of losing all breaklines.

4 CONCLUDING REMARKS

Our experience with the despike algorithm is encouraging. Excellent ground-surface models can be extracted from lidar surveys of at least some heavily forested terrain. The classification of laser survey data into ground and not-ground returns has been almost entirely automated. Such automation is desirable not only to control the costs of lidar surveys, but also to make explicit all interpretive biases and assumptions.

The classification of lidar returns—virtual deforestation—is not yet a mature art. Topics that need further work include understanding the mechanisms that cause negative blunders, designing surveys to minimize them, and researching post-processing techniques that identify them. The identification of returns from buildings, which we have barely mentioned, needs development.

A variety of VDF algorithms are currently in use, but few are well documented in the published literature. Lidar surveys of forested terrain commonly obtain imperfect results. Together, these observations suggest that the tendency of lidar survey enterprises, whether commercial or academic, to keep their algorithms proprietary and the resulting lack of dialogue has hindered development of the robust and automatic algorithms needed for lidar surveying to reach its full potential. We urge all practitioners to publish details of their post-processing algorithms. Furthermore, we see a need for comparison of algorithms on a variety of forest and terrain types. To that end, we have posted our algorithms, sample all-return data, and derived ground surface models at *http://pugetsoundlidar.org*.

Acknowledgements:

It has been a pleasure to work with our colleagues on the Puget Sound Lidar Consortium: Erik Anderson, Phyllis Mann, and David Nash (Kitsap County), Greg Berghoff (Kitsap PUD), Ken Conradi (City of Seattle), Jerry Harless (Puget Sound Regional Council), and Sam Johnson and Craig Weaver (USGS). We thank them for their efforts in obtaining funding for data acquisition, managing the acquisition contract, and evaluating the results of our VDF research. The staff at TerraPoint LLC was impressively cooperative in incorporating our VDF algorithm in their post-processing procedures. We thank Rick Blakely and Joe Means for helpful comments on this manuscript.

REFERENCES CITED

[Hansen and Vögtle, 1999] Hansen, W., and Vögtle, T., 1999. Extraktion der Geländeoberfläche aus flugzeuggetragenen Laserscanner-Aufnahmen. Photogrammetrie Fernerkkundung Geoinformation, pp. 229-236.

[Harding and Berghoff, 2000] Harding, D.J., and Berghoff, G.S., 2000. Fault scarp detection beneath dense vegetation cover: Airborne lidar mapping of the Seattle fault zone, Bainbridge Island, Washington State. Proceedings of the American Society of Photogrammetry and Remote Sensing Annual Conference, Washington, D.C., May, 2000, distributed on CD-ROM, also available at *http://pugetsoundlidar.org*

[Huising and Gomes Pereira, 1998] Huising, E.J., and Gomes Pereira, L.M., 1998. Errors and accuracy estimates of laser data acquired by various laser scanning systems for topographic applications. ISPRS Journal of Photogrammetry and Remote Sensing, v. 53, pp. 245-261.

[Kilian and others, 1996] Kilian, J., Haala, N., and Englich, M., 1996. Capture and evaluation of airborne laser scanner data. International Archives of Photogrammetry and Remote Sensing, Vienna, Austria, v. XXXI, pt. B3, pp. 383-388.

[Kraus and Pfeifer, 1998] Kraus, K., and Pfeifer, N., 1998. Determination of terrain models in wooded areas with airborne laser scanner data. ISPRS Journal of Photogrammetry and Remote Sensing, v. 53, pp. 193-203.

[Nelson and others, 1999] Nelson, A.R., Pezzopane, S.K., Bucknam, R.C., Koehler, R., Narwold, C., Kelsey, H.M., LaPrade, W.T., Wells, S.J., and Johnson, S.Y., 1999. Late Holocene surface faulting in the Seattle fault zone on Bainbridge Island, Washington (abstract). Seismological Research Letters, v. 70, p. 233.

[Pfeifer and others, 1999] Pfeifer, N., Reiter, T., Briese, C., and Rieger, W., 1999. Interpolation of high quality ground models from laser scanner data in forested areas. International Archives of Photogrammetry and Remote Sensing, La Jolla, California, USA v. XXXII, Part 3-W14, pp. 31-36, 1999.

[Schenk and others, 1999] Schenk, T., Csatho, B., and Lee, D.C., 1999. Quality control issues of airborne laser ranging data and accuracy study in an urban area. International Archives of Photogrammetry and Remote Sensing, La Jolla, California, USA, v. XXXII, Part 3-W14, pp. 101-108.

[TerraScan, 1999] TerraScan, 1999. TerraScan for microStation, user’s guide. TerraSolid Limited.