

# FILTERING STRATEGY: WORKING TOWARDS RELIABILITY

George Sithole  
Department of Geodesy, Faculty of Civil Engineering and Geosciences  
Delft University of Technology  
The Netherlands  
g.sithole@citg.tudelft.nl

Commission III, Working Group 3

**KEY WORDS:** laser scanning, LIDAR, DEM/DTM, classification, filtering

## ABSTRACT

The filtering of a laser scanner point-cloud to abstract the bald earth has been an ongoing research topic in laser altimetry. To date a number of filters have been devised for extracting DEMs from laser point-clouds. The measure of the performance of these filters is often based on tests against some reference data (rms, ratio of misclassifications vs. correct classifications, etc.) obtained by photogrammetric measurement or other means. However, measures based on such tests are only global indicators of how the filters may perform. Therefore, when applied to real life applications, based on such measures it is not possible to say with certainty how well a filter has performed. This uncertainty suggests that a method be devised to identify in a point-cloud those regions where a filter may have difficulty classifying points. This done other sources of information can be gathered to clarify the status of points in (difficult) regions. This fits in with the thinking that external sources of data, such as imagery, maps have been used in the filtering of laser scanner point-clouds. However, devising a method as suggested above requires that the reasons for the misclassification of points be first identified. When filtering a point-cloud based on spatial information alone, misclassification arises from three sources, (1) the nature and arrangement of objects and the terrain in a landscape (e.g., terrain, buildings, vegetation, etc.) (2) the characteristics of the data (resolution, outliers, data gaps, etc.) and (3) the implementation of filters. In this paper, the first two reasons for misclassification are outlined because they are common to all filtering problems, and an initial attempt at developing a method for identifying regions in a point-cloud where a filter may have trouble in classifying points is described.

## 1 INTRODUCTION

The filtering of laser scanner point-clouds to abstract the bald earth has been an ongoing research topic in laser altimetry. Filtering in the context of this paper is understood to mean the removal from a laser scanner point-cloud of those points that do not belong to the terrain. To date various filters have been developed, e.g., (Kraus & Pfeipfer 1998), (Axelsson 1999), (Petzold et. al. 1999), (Elmqvist 2001), (Vosselman & Maas 2001), (Haugerud et. al. 2001). Additional to these are proprietary filters (being used in practice), whose algorithms are not known. The performance of filters is currently measured by comparison of filter results with some reference data (rms, ratio of misclassifications vs. correct classifications, etc.). However, there are three problems with performance measures based on reference data:

- They are global indicators. The performance of filters is unified into one single measure. Unfortunately, experience has shown that this masks localized filtering errors (which can sometimes be large, although few in number).
- The performance measures are indicative of filter performance only in areas that have characteristics similar to those in the reference data (used for deriving performance measures). For example, if performance measures are derived from reference data set in a rural landscape, then in practice those measures cannot be applied to gauge filter performance in urban areas.
- Reference data is usually only available for areas that can be measured photogrammetrically or by conventional

survey. For example, in reference data, areas covered by dense vegetation are usually not sampled.

Nonetheless, filters are developed with the expectation that they will succeed. This expectation is founded on the knowledge that in most cases the characteristics of the terrain (e.g., slope, roughness, form, curvature, etc.) are bounded and that surfaces that fall outside these bounds are not terrain. Figure 1 depicts the current approach to filtering. From a monotonic function (based on a test point and its neighborhood) a decision measure is derived. Based on a pre-selected threshold for the decision measure the test point is classified as either terrain or object. As already stated this strategy works in most cases. However, it also leads to some misclassifications. Therefore, improving filtering strategy requires that the reason for misclassification be known. Two types of misclassification are possible, Type I errors and Type II errors. Type I errors being the misclassification of terrain points as object points and Type II errors being the misclassification of object points as terrain points. This is also shown in Figure 1. Within a certain band, either side of the threshold a point has a likelihood of having been misclassified (hence, the question marks in Figure 1). Knowing where and what type of misclassification has occurred can be determined using reference data. However, in normal practice this reference data will not exist. Therefore, the challenge is to detect misclassification (or the likelihood of it) where no reference data exists, or to be more precise, devise an alternative means checking the possibility of misclassification.

When reference data does not exist, an operator has to manually check the results of the filtering. However, an operator using just the point-cloud cannot achieve a 100% classification. There

will be places in the point-cloud where an operator will not be sure of the classification of points or even worse will not be able to make a classification. Therefore, three classifications are possible, "Terrain", "Object" and "Unclassified". The "Unclassified" points are those that an operator may not be able to classify with full certainty. This is in contrast with the current practice of filtering DEMs from laser scanner point-clouds, where only two classes are allowed for, "Terrain" or "Object".

The difficulty in manually classifying a point-cloud lies partly in the inherent characteristics of a point-cloud. If we can identify those characteristics of a point-cloud that lead to a difficulty to classify then we would be able to devise an alternative means of checking if classification is possible. This paper is a look at some issues related to this problem with a view to developing some framework for classification and a means of identifying regions in a point-cloud that maybe difficult to classify.

In the first section of the paper, several spatial characteristics that contribute to a difficulty in classifying a point-cloud are outlined. In the second section, the implication of the outline for future filter designs and strategies is discussed. In the final section, a trial procedure for detecting "unclassifiable" regions is described.

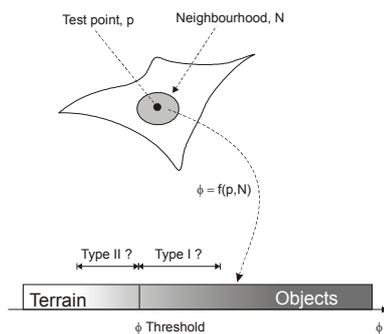


Figure 1 Current approach to filtering

## 2 DIFFICULTY TO CLASSIFY

Considering the characteristics of a point-cloud, the difficulty in classifying originates from the characteristics of the landscape and the characteristics of the laser scanner data.

### 2.1 Characteristics of Landscape

"Unclassifiable" regions due to landscape characteristics occur because of the nature and arrangement of objects and the terrain - e.g., terrain, buildings, vegetation, etc. The difficulty to classify due to landscape characteristics will be a problem in every laser scanner point-cloud without exception.

**Complexity of the terrain** - Here complexity is in reference to the form of the terrain (not roughness), in particular discontinuities (e.g., embankments, raised platforms, terraces, etc.). This characteristic especially comes into play if an operator is only able to see a portion of the point-cloud at a time, or if the terrain in a small neighborhood changes drastically in

relation to the terrain around it (discontinuity, platforms, etc.) such that it no longer fits into the general form of the terrain. Usually an operator is able to see quite a large portion of the terrain so he/she is able to appreciate the form of the terrain. This problem is demonstrated in Figure 2 (a) and 2(b).

**Complexity of objects** - While most buildings in urban areas are regular in shape (blocks, prisms, etc.) there are some that are more complex (layered roofs, platforms, etc.) and hence more difficult to classify. Another way in which objects are made complex is by the nature of objects and terrain around them. For example, a depression (e.g., a pool, excavation, etc.) near a building makes any land between the depression and the building uncertain (Figure 2d).

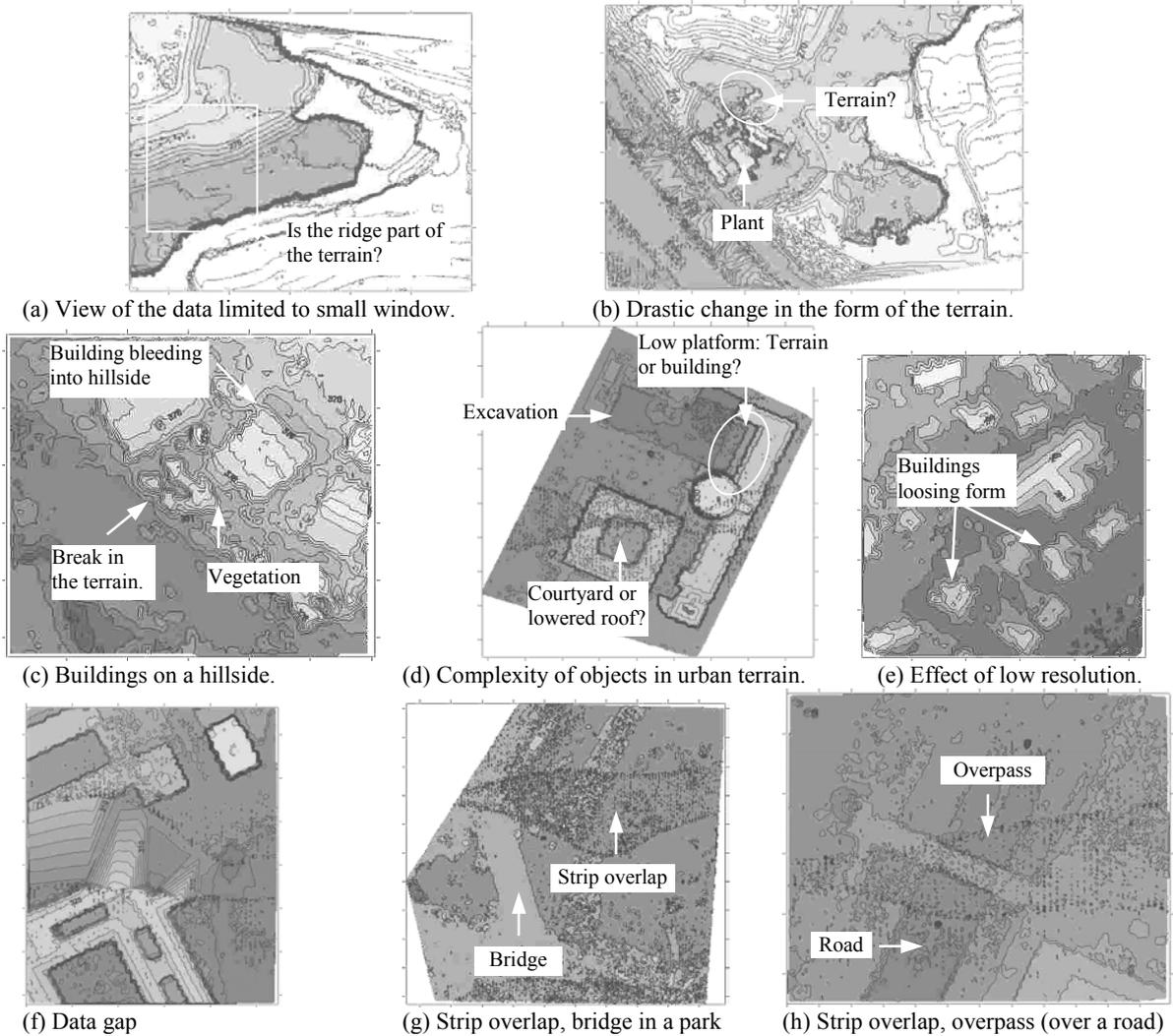
**Proximity of Object and terrain** - The closer an object is to the terrain the more difficult it becomes to distinguish it from the terrain. This separation can be lateral (in the case of sloped terrain), vertical or both. Expressed differently the separation of the surface of an object and the surface of the terrain becomes more difficult as the surfaces start to approach each other in whole or part, Figure 2(c). The ability to separate the terrain and objects is further complicated by the size of objects. The closer and larger an object is in relation to the terrain, the more difficult it is to separate it from the terrain. Therefore, the larger the area of the object surface, in relation to the area of the terrain surface, the greater the difficulty of separation.

**Zero ground returns (lack of terrain information)** - In built-up and densely vegetated areas, the number of returns from the terrain surface can approach nil. All filtering involves the comparison of a point with its neighborhood. For filtering, this comparison is meaningful only if some points in the neighborhood belong to the terrain. Zero ground returns for a neighborhood may make the result from a filter meaningless. The ability to detect this problem depends on the extent of the area for which there is no ground return. The larger the area the greater the difficulty of separation.

### 2.2 Characteristics of data

"Unclassifiable" regions due to data characteristics occur because of filtering objectives and filtering assumptions. The difficulty to classify due to data characteristics will be a problem most of the time. However, unlike those due to landscape characteristics, the difficulty of classification due to data characteristics can to a certain extent be controlled.

**Resolution of the point cloud** - The resolution of a point cloud has a direct influence on the spatial definition of objects. The less defined an object the more it starts to blend into the general character of the terrain. In Figure 2e is shown buildings on a hillside. Because of the low resolution, buildings (about 15m in length) to the bottom and right have started to lose definition and blend into the terrain. Here the proximity of object points also has an influence on the ability to discriminate between the object and the terrain. The higher the object, the easier it is to discriminate it from the terrain. As the resolution of laser



**Figure 2** Examples of “difficulty to classify”

scanner point-clouds continue to increase, this problem will become less relevant in future.

**Data gaps** - Objects that lie along or near the edge of a point-cloud are usually only partially abstracted. For an operator this lack of completeness makes any object on the edge of a point-cloud difficult to identify and hence difficult to classify. Likewise, where there are gaps in the point-cloud a similar situation to that on the edge of a point-cloud exists (Figure 2f). The usual sources of data gaps are flight-planning errors, water bodies and occlusions.

**Systematic errors** - Because of systematic distortions in a strip, the surfaces and objects abstracted in adjacent strips may not match. While an overlap has the benefit of reducing data gaps, on the other hand if there is a lack of a match between adjacent strips it creates uncertainty. In Figures 2g and 2h, the strip overlap can be seen as a band of small contours. The presence of the small contours indicates that in the overlap region the strips are offset (at least vertically). Usually this problem is not serious. It also has to be noted that filtering a strip at a time would prevent this form of problem.

**Ambiguity** - There are objects whose classification as terrain or non-terrain is not clear-cut and it usually depends on the application that will use the filtered point-cloud. Examples of such objects are bridges, and overpasses (Figure 2g and 2h). Even then, there still remains the problem of removing only those objects that are not required. For example, for whatever reason project requirements may require overpasses to be retained and bridges to be removed (both have the same characteristics!).

### 3 IMPLICATIONS FOR FILTERING

In the preceding section some factors that may lead to a difficulty to classify were outlined. Supposing we could identify in a point-cloud regions where these factors have effect then, we next have to consider its implications for future filtering strategies. A general outline of current filtering strategies is depicted in figure 3 (shaded boxes excluded). A point is selected from a point cloud. A decision criterion (function that yields the decision measure) is chosen, and a decision measure based on the spatial characteristic of a point in relationship to its neighbors is computed.

The use of alternative sources of data (e.g., aerial imagery, remote sensing imagery, maps, etc.) to support the filtering process has already been proposed Kraus & Pfeifer (1998), Ackermann (1999), although it is not known if any filters have been implemented along this line. If alternative sources of data are available then criteria derived from such data can be used to support the decision making process. However, one difficulty with multi-criteria decision-making is that the criteria considered may conflict and how these conflicts should be resolved is not always clear. With the current filtering strategy there is one aspect of conflict resolution that would have an unsatisfactory result. With the first decision criteria (based on spatial information) only two decisions are possible, “Terrain” or “Object”. But as suggested in section 1, there is a third possible decision – “Unclassifiable” because of insufficient information. If a point has been deemed unclassifiable then only external information should be used to classify it or it should be given a very low weight. But with current filtering strategies this would not be the case. Figure 3 shows how current filtering strategy could be modified to include the third possible decision. A point is first tested to determine if it is classifiable and if it is the process continues as at present. However, if the first decision criterion is not considered then, a point is classified based on criteria derived from external information alone.

Lee and Schenk (2001) propose that an intermediate process should precede all applications on laser scanner point-clouds. They propose an organization of the point-cloud. This organization is intended at mimicking the grouping of points in a point-cloud the same way human perception would. The result of the organization is the merging of points into surfaces that share common traits. This framework lends itself to the strategy envisaged in figure 3. However, the thrust here is somewhat different in that in the end, the proposed approach searches for surfaces that because of their arrangement in space have the potential of leading to a “difficult to classify”.

The strength of such a filtering strategy is that it protects against over design. With the current approach to filter design there is the danger that filters could be created to solve problems that

cannot be solved. The next section looks at an attempt to identify regions in a point-cloud that may be difficult to classify (because of landscape characteristics).

## 4 IDENTIFYING “UNCLASSIFIABLE” REGIONS

### 4.1 Problem Statement

The problem to be solved can be stated as: Determining regions in a laser scanner point-cloud where we may not normally be expected to be able to classify points as “terrain” or “object” with full certainty. Looked at differently the landscape and data characteristics noted in Sections 2.1 and 2.2 lead to the classification “unclassified”, and the aim is to find where such characteristics exist in a point-cloud. For the sake of simplicity the characteristics of Sections 2.1 and 2.2 are generalized into three groups:

- Separation – If a surface is considerably elevated above every adjacent surface, it is safe to assume that the elevated surface belongs to an object and not terrain. However, if the surface is not above every adjacent surface then there is the likelihood that it may belong to “object”, “terrain”, or maybe “difficult to classify”. The classification in this case depends on how close (vertically) the surface is to other surfaces and its planimetric position in relation to other adjacent surfaces. Finally, if a surface lies below every other surface adjacent to it then there is the possibility that it belongs to “terrain” or “difficult to classify”. In this case whether a surface belongs to “terrain” depends mainly on its surface area. By testing surfaces in this manner those surfaces that obviously belong to “object” or “terrain” are determined and what remains is treated as “difficult to classify”.
- Contradiction – This situation refers to objects that are linked to the terrain but are at the same time also above it. Objects that lead to this are those described in Section 2 under Ambiguity.
- Data scarcity – Gaps in the data.

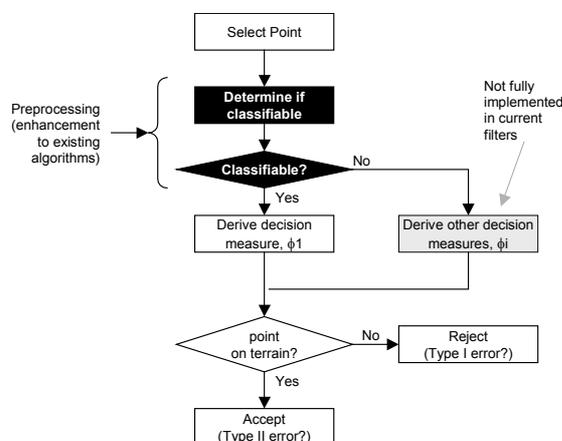


Figure 3 Revised filtering strategy (shaded figures represent enhancements).

Here zero ground return, data resolution, and systematic errors are not considered. One possible way of identifying uncertain regions is to apply a filter on a point cloud using different parameter values. The areas whose classification changes with a variation of the parameter values are treated as uncertain. Gooch and Chandler (1999) use such a scheme in the prediction of failure in automatic DEM generation. However, some features (such as large buildings) are invariant to changes in filter parameters. Therefore, another approach is tried here, and this is explained in the next section.

### 4.2 Implementation

The implementation discussed in this section is only limited to the identification of “terrain” “difficult to classify” and “object” based on separation.

Points (that share common characteristics) first have to be grouped into surfaces. Later the surfaces can be compared with each other to see if their separation leads to a classification of “object”, “terrain” or “unclassified”. The implementation here is based on gathering information for surfaces from profiles and aggregating that information to resolve the relationship of surfaces to each other. Figure 4 shows the procedure for detecting regions where surfaces may be visually inseparable.

1. A point-cloud is (Figure 4(a). Here approx. 17000 points from a City Landscape) is partitioned into rows and columns along the direction of the x and y, axes. This partitioning scheme is later used to segment the surfaces and to compute the surface scores.
2. Profiles are generated for every row (y) and column (x). In every profile points are sorted and then connected if they are within a certain height, distance, etc., of each other. Therefore, every profile will yield strings of points, and where there are discontinuities the strings will be disjoint. Figure 4(b). In essence the strings are samples of surfaces. Later by comparing how the strings of a surface relate to the strings of adjacent surfaces a general impression will be formed of how one surface relates to another surface.
3. By aggregating the strings in the x and y directions surfaces are generated. Figure 4(c). Computation and assignment of surface scores is done in step 5.
4. In this operation two adjacent strings in a profile are compared at a time. As mentioned already where there are discontinuities the strings will be disjoint. Whether the discontinuity is large enough to make it difficult to visually separate the eventual surfaces is determined by a score function (given in Equation 1, and graphed in Figure 5).

The greater the separation of strings, the greater the score. The score function works on two adjacent strings, and a score (0 to 1) is assigned to the end of the higher string, Figure 5. The end of the lowest string is given a “no score” indicating that based on the comparison no assessment can be made about the string end. This is essentially the coding of separation as described in section 4.1. In this manner a score is assigned to the end of every string, Figure 4(d). The parameters ‘a’ and ‘b’, of Equation 1 are selected so that string ends separated in height by an amount less than ‘a’ receive a zero score and those greater than an amount ‘b’ receive a score of 1. Height differences between ‘a’ and ‘b’ attract a score graded between 0 and 1. Values of 0.5m and 2m for parameters ‘a’ and ‘b’ respectively proved workable.

$$\phi(dh) = \begin{cases} \text{no score} & dh < 0 \\ 0 & 0 \leq dh < a \\ \frac{(dh - a)}{(b - a)} & a \leq dh < b \\ 1 & dh \geq b \end{cases} \quad (1)$$

5. Strings that lie on the same surface should have one or more points in common. This knowledge is used to segment the point-cloud into surface segments. Because the strings on a surface will have different scores, to obtain an overall score for the surface will require an aggregation of the scores for all strings. This is done by computing the weighted mean of the scores for all strings weighted by their lengths. However, before this can be done an overall score has to be determined for each string. The overall score for a string is based on the scores at the string ends. If a string has numeric (0...1) scores at both ends then the

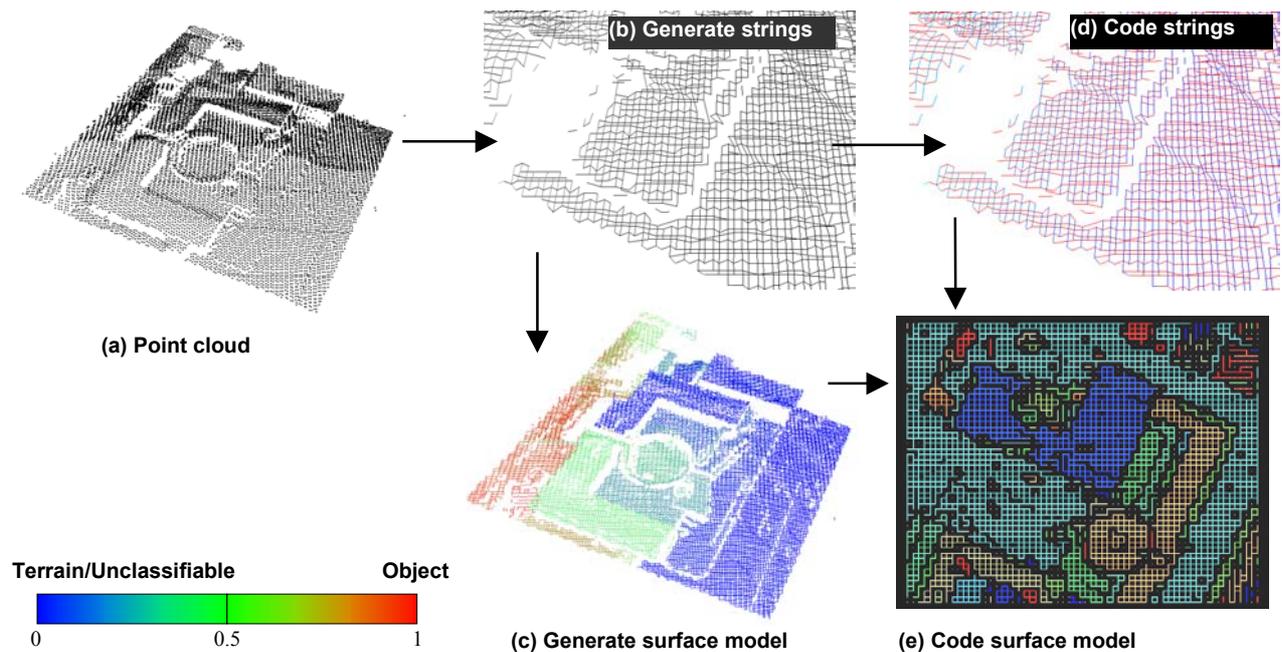


Figure 4 Coding separation

## 5 CONCLUSION

In this paper it has been proposed that in a preprocessing step a point cloud should be segmented and that the results of this should drive the filtering. Further still it is argued that this preprocessing step is essential if external information is to be used in a filtering procedure, to clarify or validate the classification of regions that cannot be classified on positional information alone. Some of the characteristics of the landscape and data that lead to a difficulty to classify laser point-clouds have been highlighted. These characteristics are then used to search for regions in the point cloud that cannot be classified with certainty. An initial attempt at developing an application to segment such regions has been demonstrated. However, this application still requires much improvement and there are also other aspects such as identifying contradictions, and data gaps, data resolution, etc., that will have to be treated.

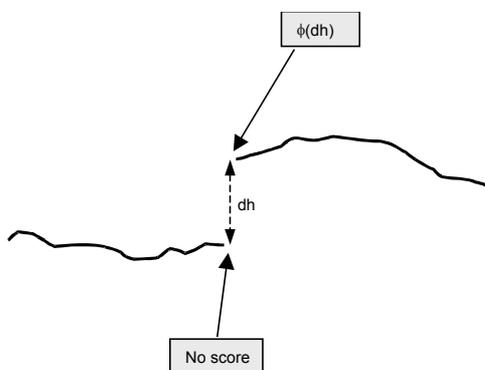
## REFERENCES

- Axelsson P., 1999: "Processing of laser scanner data - algorithms and applications". ISPRS JPRS, 54 (1999). pp. 138 - 147.
- Ackemann F., 1999: "Airborne laser scanning - present status and future expectations". ISPRS JPRS, 54 (1999). pp. 64 -67.
- Elmqvist, M., 2001: "Ground Estimation of Laser Radar Data using Active Shape Models". Paper presented at the OEEPE workshop on airborne laserscanning and interferometric SAR for detailed digital elevation models 1-3 March 2001, paper 5 (8 pages). Royal Institute of Technology Department of Geodesy and Photogrammetry 100 Stockholm, Sweden.
- Gooch, M., Chandler, J., 1999; "Failure Prediction in Automatically Generated Digital Elevation Models". 4th International Conference on GeoComputation, Virginia, USA, 1999, CDRom.
- Haugerud R.A., Harding D.J., 2001: "Some algorithms for virtual deforestation (VDF) of LIDAR topographic survey data". IAPRS, Vol. 34-3/W4. pp. 211-218.
- Lee, I. and Schenk T., 2001: "3D perceptual organization of laser altimetry data". IAPRS, Vol. 34-3/W4. pp. 57 - 65.
- Kraus, K., Pfeifer, N., 1998: "Determination of terrain models in wooded areas with airborne laser scanner data. ISPRS JPRS. Vol. 53, pp. 193-203.
- Petzold B.; Reiss P.; Stossel W.; 1999: "Laser scanning - surveying and mapping agencies are using a new technique for the derivation of digital terrain models." ISPRS JPRS. Vol. 54 (1999). pp. 95 - 104.
- Vosselman, G., 2000: "Slope based filtering of laser altimetry data". IAPRS, Vol. 33/B3, pages 935-942.

mean is taken. If a string has a numeric score at one end and a "no score" at the other end, the numeric value is used. If a string has a "no score" at both ends, then zero is used. In this way an overall score is obtained for each surface. Reinterpreted the score indicates if a surface is "terrain/unclassified" (values close to 0) or "object" (values close to 1). Seen in Figure 4(e), the algorithm has identified the roofs as "object" regions (here shown as red) that should pose no problems for filtering. This leaves the bluer regions (values close to 0) that represent areas that maybe terrain or "difficult to classify". In the first instance "terrain" and "difficult to classify" regions can be discriminated by an examination of surface areas. The larger the surface area the more likely that the surface is terrain. Once terrain (large areas) areas have been determined in this way, then, remaining terrain/unclassified regions can be tested against the already determined terrain surfaces. If a surface is below already determined terrain then very likely it too is terrain. Further to this low points in the point-cloud can be determined. If these points coincide with surfaces determined as terrain, then this would serve as confirmation.

6. Once surfaces have been coded as "terrain", "object" or "difficult to classify" then every point in the point-cloud also has to be coded. This is done by assigning to every point the score of its corresponding surface.

Once surfaces have been coded the search for ambiguities also becomes possible by examination of the string scores. For example a terrain surface should not have strings whose scores are close to 1. In here lies the strength of the procedure in that the local context (string scores) are used to obtain a global context (surface scores) for the landscape. The above procedure is still very much under development but is shown here to demonstrate a possible way for identifying regions that might be difficult to classify. Moreover, the detection of contradictions and the determination of the influence of gaps on strings should provide more realistic results. Furthermore, the manner in which strings were generated is not yet ideal because it is based solely on height difference. Future, implementations will use the area of the different segments and second returns to strengthen the classification process.



**Figure 5** Assigning scores to string ends (strings separated in height).