An adaptive surface filter for airborne laser scanning point clouds by means of regularization and bending energy

Han Hu [a], Yulin Ding [a,b,*], Qing Zhu [a,c], Bo Wu [d], Hui Lin [b], Zhiqiang Du [a], Yeting Zhang [a], Yunsheng Zhang [e]

[a] State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, P.O. Box C310, 129 Luoyu Road, Wuhan, Hubei 430079, PR China
[b] Institute of Space and Earth Information Science, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
[c] Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu, Sichuan, PR China
[d] Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
[e] School of Geosciences and Info-Physics, Central South University, Changsha, Hunan, PR China

ABSTRACT

The filtering of point clouds is a ubiquitous task in the processing of airborne laser scanning (ALS) data; however, such filtering processes are difficult because of the complex configuration of the terrain features. The classical filtering algorithms rely on the cautious tuning of parameters to handle various landforms. To address the challenge posed by the bundling of different terrain features into a single dataset and to surmount the sensitivity of the parameters, in this study, we propose an adaptive surface filter (ASF) for the classification of ALS point clouds. Based on the principle that the threshold should vary in accordance to the terrain smoothness, the ASF embeds bending energy, which quantitatively depicts the local terrain structure to self-adapt the filter threshold automatically. The ASF employs a step factor to control the data pyramid scheme in which the processing window sizes are reduced progressively, and the ASF gradually interpolates thin plate spline surfaces toward the ground with regularization to handle noise. Using the progressive densification strategy, regularization and self-adaption, both performance improvement and resilience to parameter tuning are achieved. When tested against the benchmark datasets provided by ISPRS, the ASF performs the best in comparison with all other filtering methods, yielding an average total error of 2.85% when optimized and 3.67% when using the same parameter set.

© 2014 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS) Published by Elsevier B.V. All rights reserved.

1. Introduction

Airborne Laser Scanning (ALS) systems present promising alternatives to traditional airborne photogrammetry (Wehr and Lohr, 1999; Zhang et al., 2003; Vosselman and Maas, 2010) in the generation of Digital Elevation Models (DEMs), surface reconstructions, environmental surveys and many other applications (Haala and Kada, 2010; Mongus and Žalik, 2012; Hauglin et al., 2013). Because the raw data consist of a combination of the significant number of points returned from diverse terrain features (e.g., ground, buildings, vegetation and other objects), before being adapted to many other applications, ground and non-ground points must be separated first. This process is referred to as ALS points filtering (Meng et al., 2010). The filtering of ALS data is a particularly demanding task because the data normally cover large areas and various types of surface objects. Previous related publications indicate that ALS data filtering is an extraordinarily difficult task (Sithole and Vosselman, 2004) and is still currently actively under investigation (Mongus and Žalik, 2012; Véga et al., 2012; Chen et al., 2013; Li, 2013; Maguya et al., 2013; Pingel et al., 2013; Zhang and Lin, 2013). Because more ALS datasets are becoming readily available, an innovative ALS filtering algorithm with improved and stable performance is urgently needed to reduce the amount of time-consuming manual editing (Flood, 2001; Chen et al., 2013).

1.1. Filtering strategies

Various types of filtering methods have been proposed. Based on the filter strategies, these algorithms can be grouped into four major categories (Liu, 2008; Meng et al., 2010): interpolation-based (Kraus and Pfeifer, 1998; Axelsson, 2000; Evans and Hudak, 2007; Mongus and Žalik, 2012; Chen et al., 2013), slope-based

(Vosselman, 2000; Sithole, 2001), morphological-based (Zhang et al., 2003; Chen et al., 2007; Li, 2013; Pingel et al., 2013) and segmentation/cluster-based filters (Filin, 2002; Sithole and Vosselman, 2005; Zhang and Lin, 2013). For the interpolation-based methods, the initial ground points are selected and then densified iteratively to create a provisional surface that gradually approaches the final ground surface. The slope-based methods are based on the assumption that the gradient of the ground is obviously smoother than that of non-ground objects (Sithole, 2001), and the threshold to distinguish ground from non-ground points is determined by a monotonically increasing kernel function (Vosselman, 2000). For the morphological-based methods, the mathematical morphology operations, e.g., dilation and erosion, are exploited to process the Digital Surface Model (DSM) (Zhang et al., 2003), and the non-ground objects can be removed by using a combination of the basic operations. The methods in the last category generally cluster the dataset in the feature space into some segments, for which normal vector and elevation differences in the neighborhood are two appropriate measurements (Filin, 2002; Zhang and Lin, 2013). Subsequently, the premise that points in the same cluster should share the same label can be used to enhance the classification.

Sithole and Vosselman (2004) presented an experimental comparison of the performance of eight filtering algorithms. The authors concluded that the interpolation-based filters often outperform the other methods in the handling of complex terrain because the sophisticated interpolation methods can partially handle various terrain features. Therefore, the interpolation-based approach is exploited in this study. In the following subsection, we do not provide an exhaustive review of all of these methods but instead highlight only the interpolation-based filters that are directly relevant to our work in the next subsection.

1.2. Interpolation-based filters

The linear prediction approach presented by Kraus and Pfeifer (1998) was an early investigation of an interpolation-based filter used to create a DEM in a wooded area. In the interpolation procedure, a weight, which ranges from 0 to 1.0, is assigned to each point. Starting with a grid surface interpolated with identity weights, the weights are determined by the residual between the elevation of the point and the interpolated surface. A dual truncated decreasing function is adopted, so points with residuals smaller than a lower bound are awarded maximum weights and those higher than an upper bound are penalized with zero weights, which will not contribute to the ground surface. In this manner, the weights and ground surfaces are both iteratively refined. Pfeifer et al. (2001) extended the method to a hierarchic scheme with a data pyramid to accelerate the filtering process. In addition, because the coarse level grids (top-level) in the data pyramid are more likely to be ground points, as they are the local minimum in a larger window, the hierarchical pyramid scheme can offer a more robust ground surface estimation. The data pyramid is commonly built in a quad-tree structure, in which a node in the upper level is linked to four nodes in the lower level, but we have noticed that making the pyramid move slowly to the bottom level will provide more accurate results (cf. Section 2.2). The iteration strategy adopted in the method is based on refining the weight assignment. However, other methods that iterate with the densification of ground points have exhibited better performances. Axelsson (2000) provided a groundbreaking report on ALS filtering based on a Triangulated Irregular Network (TIN), also referred to as Progressive TIN Densification (PTD) (Zhang and Lin, 2013). Although the original progressive TIN surface filter has provided promising work with excellent performance (Sithole and Vosselman, 2004), it turns out that optimization details, which were kept proprietary in the original PTD approach by Axelsson (2000), have great im-

pacts on algorithm accuracies (Zhang and Lin, 2013). Most investigations tend to use a gridded surface with more sophisticated interpolation methods, which can also achieve comparable results (Mongus and Žalik, 2012; Chen et al., 2013).

Recently, an interpolation-based method using the Thin Plate Spline (TPS) approach as the interpolant was demonstrated to be experimentally more suitable for ALS filtering compared with other interpolation techniques, e.g., Kriging, Inverse Distance Weighting (IDW) and TIN, according to the work by Evans and Hudak (2007). The authors introduced a multiscale curvature classifying (MCC) algorithm for the filtering of ALS data. In contrast to representing the ground surface with TIN (Axelsson, 2000), MCC employs regular gridded DEM. After selecting the initial ground points, these points were used to interpolate a raster surface with TPS for the first scale, and then, unclassified points were tested against the average elevation of the 3×3 neighbors in the DEM similar to the work by Haugerud and Harding (2001). The points were classified as ground, if the elevation difference was less than a given threshold. The process was repeated until no more points were added into ground points. Then, the process moved on to the next resolution. For a larger resolution, a scale gain (0.1 m) was added to the curvature threshold to address the effect of changes in slope (Chen et al., 2013). Three scales were used in total, and they were determined as *0.5w*, *w*, and *1.5w*. The curvature threshold was *t*, *t* + 0.1, and *t* + 0.2, respectively, where *w* and *t* are the user-defined initial scale and threshold, respectively. Chen et al. (2013) proposed a similar method with multi-scale TPS interpolation, and by performing tests against the benchmark dataset supplied by the ISPRS Commission, the authors demonstrated the outstanding performance of TPS interpolation in ALS filtering. Although the interpolation-based method with TPS described above (Evans and Hudak, 2007; Chen et al., 2013) adopted three levels of interpolated scale, the points were not prepared in the pyramid structure. At each scale, all points were tested against a curvature threshold instead of processing in the coarse-to-fine sequence. Furthermore, Mongus and Žalik (2012) presented a TPS interpolation-based algorithm without parameter tuning. After building a data pyramid of the point clouds, a surface is interpolated iteratively from the coarsest level toward the finest level. The removal of parameter tuning is achieved based on the statistical information of the elevation residuals between points and DEM. The benchmark tests demonstrated that the method exceeded the performance of the software standard, even with automatically determined parameters.

1.3. The ASF approach

As described above, most of the previous studies have determined the filter threshold using elevation information only, and the threshold remained the same for a single dataset, even with various terrain features. Most algorithms achieve arguably excellent performance when applied to consistent and plain areas, but the filters remain problematic when faced with complex shapes/configurations and significant discontinuities (Sithole and Vosselman, 2004). Multiple factors can account for the problem, with the major factor likely being the complexity of the landform. In fact, the challenge of handling different terrain features bundled into a single dataset has already been explored in previous works (Sithole and Vosselman, 2004; Zhang and Lin, 2013). For example, in an actual filtering problem, the region may contain low objects or vegetation on flat surfaces in addition to sharp ridges or scarps on rough surfaces. The varying features must be handled by different filtering thresholds. A small threshold should be assigned to de-spike the low objects, and a larger threshold should be applied to retain the ground points on the tops of ridges or on the edges of scarps. Sithole and Vosselman (2004) proposed the

use of additional contexts to address this problem. Two types of implementations have been attempted, including (a) segmenting the point clouds into a smooth surface in the preprocessing stage and then classifying the segments (Filin, 2002; Zhang and Lin, 2013) and (b) using elevations in a 3×3 neighborhood during the filtering process rather than only one grid (Haugerud and Harding, 2001; Evans and Hudak, 2007; Chen et al., 2013).

To improve the filter performance and robustness with respect to complex scenes, in this study, we propose a novel adaptive surface filter (ASF) for ALS data processing. Compared with previous works (Evans and Hudak, 2007; Mongus and Žalik, 2012; Chen et al., 2013), the proposed method has the following three innovative aspects. (1) The ASF controls the initialization of the levels for the data pyramid using a step factor instead of the quad-tree structure, which interpolates the raster surface to gradually approach the ground surface and improve the robustness of the algorithm (cf. Section 2.2). (2) Regularization is used to overcome noise during interpolating TPS surfaces using progressively densified ground points to the final ground (cf. Section 2.3). (3) A bending energy function that explicitly depicts the surface smoothness is used and embedded in the algorithm for adaptive threshold determination (cf. Section 2.4). The raster surface and the bending energy are integrated to label the remaining unclassified points as either ground or non-ground.

The rest of the paper is structured as follows. Section 2 provides the ASF algorithm details, including the preprocessing and filtering algorithm. Furthermore, the implementations of step factor, regularization and bending energy are also presented in this section. Then, the performance is evaluated qualitatively and quantitatively against benchmark datasets provided by ISPRS (http://www.itc.nl/isprswgIII-3/filtertest) using the evaluation paradigm put forth by Sithole and Vosselman (2004) in Section 3. This section also analyzes the effects of the parameters used in our algorithm. Conclusions are then presented in the last section.

2. Methods

2.1. The adaptive surface filtering method

Three contributions of the implementation details with respect to the interpolation-based method using TPS distinguish the ASF method from its predecessors (Evans and Hudak, 2007; Mongus and Žalik, 2012; Chen et al., 2013) in terms of its increased robustness and stability when handling complex terrains and outperforming other filters. First, we use a step factor to control the window size between adjacent pyramid levels instead of a quad-tree structure. Therefore, the provisional surface will move slowly but more robustly to the final ground. Second, regularization is used in the TPS interpolation to relieve the impacts from noise points in the iterative procedure. The effect of regularization is adjusted by a parameter λ, which controls the weights between the fitness of the data and the smoothness of the surface (Terzopoulos, 1988). Third, the bending energy, generated as the by-product of the TPS interpolation, is exploited to enable the threshold to adaptively change in accordance with the terrain variations. The bending energy can be considered a scalar, which is relevant to the structure of the local neighbors, as shown in Fig. 1. With the local descriptor of terrain integrated into the threshold determination, the critical problems faced by all surface filters, such as complex scenes, large scarps, and breaklines, can be alleviated. With these enhancements, the algorithm is resilient with respect to parameter tuning, resulting in more stable outcomes. The entire workflow consists of two parts, i.e., preprocessing and filtering, as shown in Fig. 2.

Due to the ability of laser pulses to penetrate vegetation, several returns may be recorded in a single laser emission. It is only possible for the final return of multiple returns to be a ground point. As a result, only the final returns are considered to be ground candidates. The points that have only one return are also reserved for successive processing. In addition, the data used in this study only comprise points from the final single return.

Based on the local minimum constraint, the lowest points in a local square window are commonly used as the initial ground points because of their simplicity and efficiency (Chen et al., 2013). Unfortunately, as a consequence of the multi-path reflex and other errors, low outliers will cause incorrect initial ground control points and must be eliminated beforehand (Mongus and Žalik, 2012) or handled during the filtering process (Haugerud and Harding, 2001). Because these outliers are rare and usually reside far from other points, the low outliers can be simply removed by an outlier filtering technique for point clouds, such as radius removal or statistical removal filters, which are publicly available (Rusu and Cousins, 2011). These outlier filters are general-purpose methods to remove isolated points, e.g., birds, power lines and occasionally solitary ground points. The radius removal filter will remove the points whose number of neighbor points in the range of a user-defined radius is less than a certain number. In most cases, the radius removal filter is used in our study because of its flexibility when handling various situations, ranging from no outliers to clamped outliers. The radius search can be efficiently implemented based on a k-d tree (Muja and Lowe, 2009) with an algorithm complexity of $O(\log n)$. The search radius is chosen according to the point density. Based on the assumption that the points are evenly distributed in the area, the number of points lying within the search circle in the planar direction is proportional to the area of the circle. Therefore, a larger radius will preserve more points, and a large expected number of points will remove more points. The outliers removed here are not considered in the iterative filtering procedure but will be reclassified as either ground or non-ground during post-processing.

After removing the low outliers, the local minimum points are confirmed as correct ground points. Then, the data pyramids can be built from the remaining points. Instead of the bottom-up fashion utilized in the work by (Mongus and Žalik, 2012), we use a top-down approach because the step factor (denoted as s) between each level in the pyramid is not two. As a result, there is no rigorous quad-tree structure. We choose the lowest point in a moving window for the corresponding pyramid level. The window size for the top level (denoted as max_window) is determined by the size of the largest object in the area. Generally, max_window = 30 m is used in this study. In addition, the window size for the bottom level is related to the point density. For intermediate levels, the window size is determined by s, which will be detailed in Section 2.2.

The filtering algorithm consists of two iterations, the outer and the inner, as shown in Fig. 2. For the outer iteration, all levels in the data pyramids are traversed, whereas for the inner iteration, the unclassified points are tested against the surface interpolated with the ground points. For each level/scale in the pyramid, we introduce a parameter (denoted as $scale\_gain$) to the filter threshold, similar to the work of Evans and Hudak (2007). Furthermore, to exploit the bending energy to adaptively change the threshold, bending energy must be transformed into a threshold compensation (denoted as $bend\_gain$) and amended with a binary indicator (denoted as $bend\_mask$), which will be explained in Section 2.4. For detailed calculations of the TPS surface and the bending energy, please refer to the work of Elonen (2005). The filtering procedure of the ASF is as follows.

(1) Create a vector that records the ground points (denoted as G) and another vector that records the unclassified points (denoted as U).
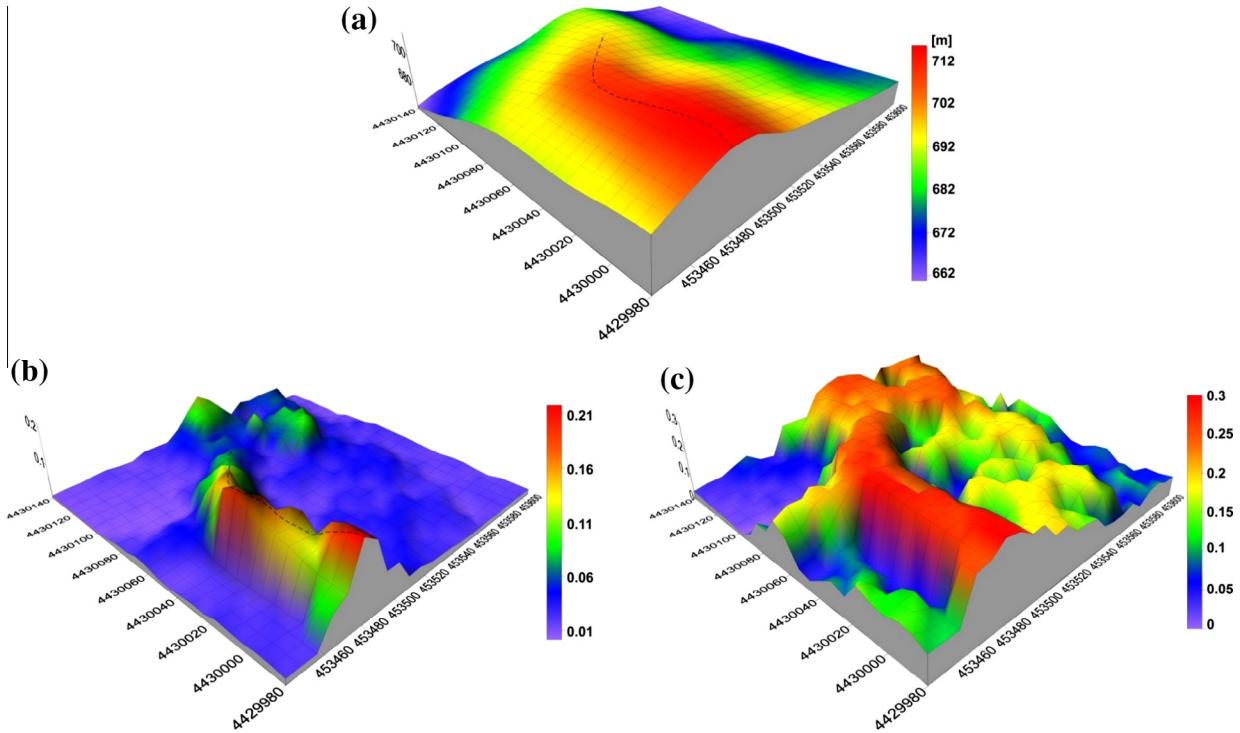
Fig. 1. Descriptions of the bending energy and the transformed compensation value generated during interpolation of the corresponding raster surface. (a) The interpolated raster surface, (b) the generated bending energy as a by-product of the TPS interpolation and (c) the transformed *bend_gain* by piece-wise linear interpolation from the bending energy raster given a upper bound of 0.3 m. The dashed curves in (a) and (b) represent the same ridge.

(2) Initialize *G* with the points in the top level of the pyramid and *U* with the points in the next level.

(3) For each level of points in the data pyramid, perform the following filter processing iteratively:

  i. Calculate the regularization parameter λ and *scale_gain* based on the pyramid level. Parameter λ increases linearly from 0 to the maximum value when progressing from the top to the bottom (*cf.* Section 2.2). In contrast, the *scale_gain* decreases from the maximum value to 0, which is similar to previous works (Evans and Hudak, 2007; Chen et al., 2013). The maximum values of λ and *scale_gain* are constantly determined by 0.5 *m* and 0.3 *m*, respectively.

  ii. Create a raster DEM at the corresponding scale using regularized TPS interpolation from points in *G*. Because of the border effects, there might be no points around the grids on the border, in this case, we detect these grids and mark them as no data. In addition, the by-products of the TPS interpolation procedure are a raster denoting the bending energy for each grid and a raster for *bend_mask*.

  iii. Transform the bending energy raster into *bend_gain* using piece-wise linear interpolation over the range from 0 to an upper bound, as depicted in Fig. 1c.

  iv. Filter the points in *U* based on the threshold z_t and the elevations of the 3×3 neighbor cells in the DEM. The threshold is jointly determined using the following Eq. (1):

$$z_t = t + scale\_gain + bend\_gain \times bend\_mask \qquad (1)$$

where *t* is a user-defined parameter; in this study, values from 0.1 m to 0.5 m are used. If the point elevation does not exceed 5 of 9 cell elevations plus z_t, then it is assigned to *G*, which is inspired from the work by Chen et al. (2013).

  v. Update *G* and *U*, and check the terminal criteria. If the new ground points are less than a certain amount, insert the points in the next level into *U* and proceed with the filtering. Otherwise, repeat (i) to (iv).

(4) Filter the points considered to be outliers during preprocessing with the same z_t defined in Eq. (1). However, another criterion for a point to be classified as ground is considered. The elevation must exceed the DEM elevation minus 3 m (Haugerud and Harding, 2001). Otherwise, it is considered as a low outlier and assigned into *U*.

(5) Label the points in *G* as ground and the points that remain in *U* as non-ground.

2.2. Step factor for constructing the pyramid

The step factor (*s*) controls the granularity between adjacent levels in the pyramid. When *s* = 2 is used, the pyramid has a quad-tree structure exactly, and if *s* approaches infinity, the coarse-to-fine strategy is entirely abandoned. The interval window sizes are multiplied by *s* from the level below. From the top level, all of the points are arranged into a regular grid at the cell resolution of the corresponding window size. Each grid cell may include a series of points, and only the lowest points in each cell are retained for this level before their removal from the dataset and proceeding to the next level. The process repeats to the bottom level. *s* = 1.2 is used in this study rather than the value of two, which is common among all other studies of data pyramid construction (Pfeifer et al., 2001; Mongus and Žalik, 2012). The reason for the progressive movement toward the bottom level is that if the value two is used, the number of points in the next level is three times greater in comparison with an increase of approximately 50% when *s* = 1.2 is chosen. If more points are tested against a relatively coarser level

```
                              ┌──────────────────┐
                              │    ALS data      │
                              └──────────────────┘
                                       │
                              ┌──────────────────────┐
                              │ Select the last returns │
                              └──────────────────────┘
                                       │
  Preprocessing               ┌───────────────────────────────┐
                              │ Remove low outliers/ negative blunders │
                              └───────────────────────────────┘
                                       │
                              ┌─────────────────────────────────────┐
                              │ Construct data pyramids with small step factor │
                              └─────────────────────────────────────┘
                                       │
                              ┌──────────────────────────────────┐
                              │ Select initial ground and unclassified points │
                              └──────────────────────────────────┘
                                       │
                              ┌──────────────────────────────────┐
                              │ Interpolate a raster surface regularized TPS │◄──┐
                              └──────────────────────────────────┘             │
                                       │                                       │
                              ┌──────────────────────────────────┐             │
                              │ Calculate bending energy for each grid │        │
                              └──────────────────────────────────┘             │
  ASF                          │                          ┌──────────────────┐ │
  Filtering            ┌────────────────────────────────┐ │ Update ground points│ │
                      │ Filter unclassified points with adaptive threshold │ └──────────────────┘ │
                      └────────────────────────────────┘             ▲         │
                                       │                             │         │
                        ◇ Ratio of new ground points < 0.01%? ◇──No──┘         │
                                       │ Yes          ┌──────────────────────┐ │
                                       │              │ Update unclassified points │─┘
                                       │              └──────────────────────┘
                        ◇ Bottom level? ◇──No──────────────┘
                                       │ Yes
                              ┌──────────────────────────────────┐
                              │ Filter outlier points removed in preprocessing │
                              └──────────────────────────────────┘
                                       │
                      ┌────────────────┴────────────────┐
              ┌──────────────┐                  ┌──────────────────────┐
              │    DEM       │                  │ Classified point cloud │
              └──────────────┘                  └──────────────────────┘
```
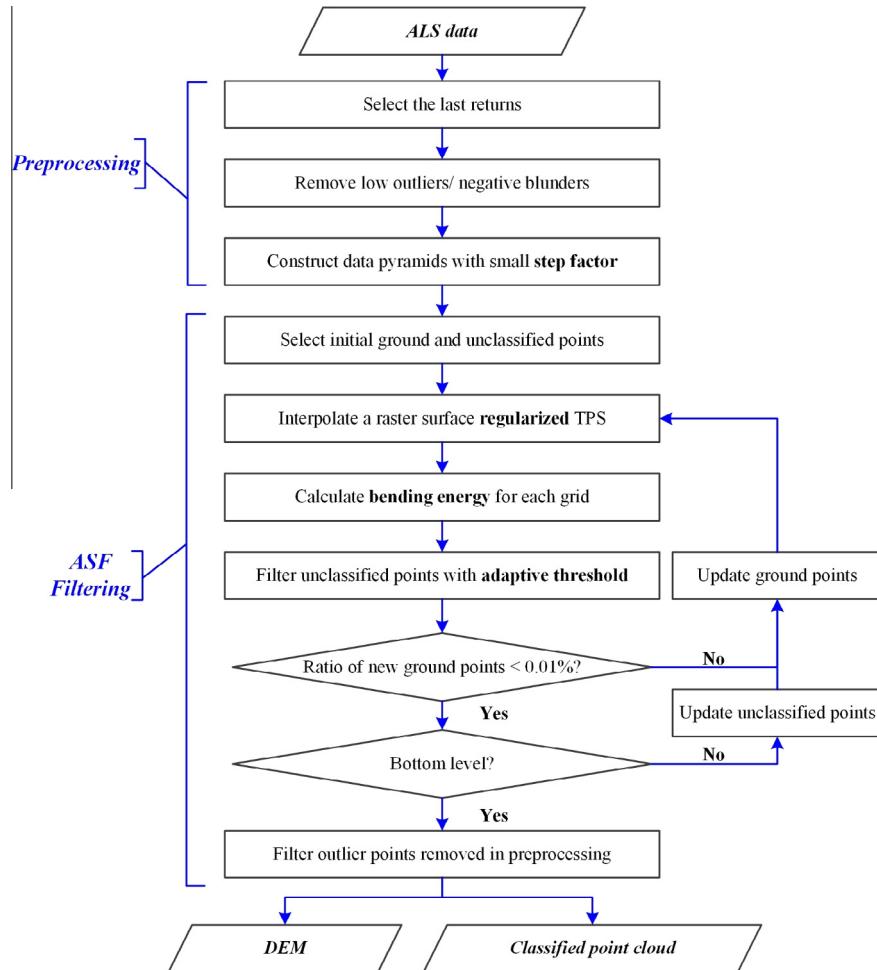
Fig. 2. Flowchart of the ASF to produce classified point clouds and DEM from the unclassified ALS data.

of surface in a single iteration, the filter process will become more inclined to misclassification. In addition, any misclassifications in the procedure are not only irreversible but also magnified in the following iterations. Furthermore, with a small step factor, the elevation difference between each level is slowly increased and thus loses more ground details.

2.3. Regularization for noise handling

In 3-D space, TPS is the solution to the following regularization problem, which is used to find a continuous and smooth surface (Szeliski, 2011):

$$\varepsilon = \sum_{i=1}^{n}(z_i - f(x_i, y_i))^2 + \lambda \int f_{xx}^2(x,y) + 2f_{xy}^2(x,y) + f_{yy}^2(x,y)\,dxdy \quad (2)$$

where the former part, $\varepsilon_1 = \sum_{i=1}^{n}(z_i - f(x_i, y_i))^2$, is referred to as the data term (Szeliski, 2011), which measures the fitness of the surface and the control points, and the latter portion, $\varepsilon_2 = \int f_{xx}^2(x,y) + 2f_{xy}^2(x,y) + f_{yy}^2(x,y)\,dxdy$, which is normally called the smoothness penalty in an energy minimization problem (Morse et al., 2005; Szeliski, 2011). In the case of 3-D space, if the regularization parameter $\lambda = 0$, no regularization is imposed, and the surface will pass exactly through all of the given points, which results in a perfect data fitness ($\varepsilon_1 = 0$). No regularization was used in all the previous TPS-based interpolation filters (Evans and Hudak, 2007; Mongus and Žalik, 2012; Chen et al., 2013). In contrast, as λ approaches infinity (over regularized), the surface reduces to a least-square

fitted plane, which is a completely smooth surface in theory ($\varepsilon_2 = 0$) (Elonen, 2005). In addition, intermediate values of λ will produce a compromise between the fitness of the data and the smoothness of the surface.

When noise clutters the ground points during filtering, the optimization obtained from regularization is crucial for deriving a ground surface that is resistant to errors. As shown in Fig. 3, some object points will not be rejected, such as the shaded area in Fig. 3b and c. If not regularized, the ground surface will pass exactly through each ground point, as indicated by the dashed line in Fig. 3b and in the following iterations, the ground points may climb up, accepting all of the object points in the same spot as the final dotted surface in Fig. 3b. In contrast, although noise still exists after the first iteration, the interpolated surface will not pass exactly through all of the points and generate a regularized surface. In this manner, the subsequent iterations will not magnify the errors.

However, the advantage of regularization comes at the cost of the loss of some terrain details, as expressed by the final surface with the slope where the ground surface will not exactly pass through the ground points as depicted in Fig. 3c. Thus, λ must be selected scrupulously when compromising between resilience to noise and preserving detailed information. When applied to interpolate a ground surface in the filtering algorithm, an intuitive consideration in determining λ is that it should be related with the amount of noise in the ground points and more specifically, it is a positive correlation. In the filtering procedure, the initial ground points are assured to be ground points, and noises increase with
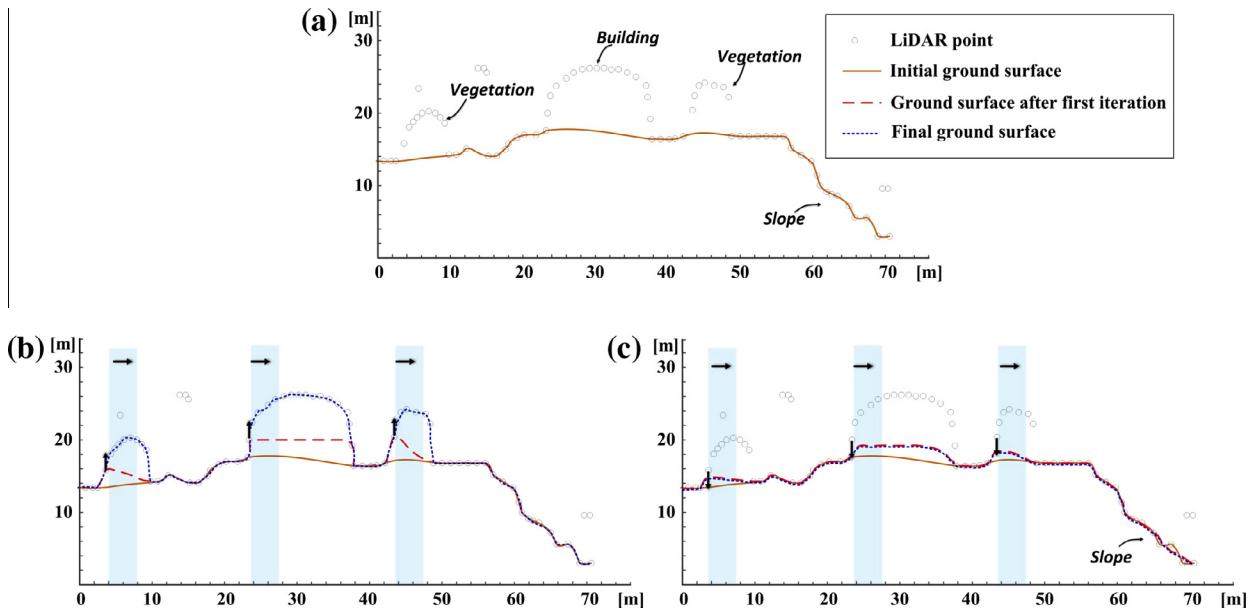
Fig. 3. Effects of the regularization parameter in resisting magnification of noise points during the iterative filter process. (a) The ground surface interpolated with initial ground points, (b) the iterative filter process without regularization ($\lambda = 0$) and (c) the iterative filter process with proper regularization. The upward arrows in (b) refer to the climbing-up effects of the noise points without regularization, and the downward arrows in (c) denote the regularized effects on the ground surface that facilitate avoidance of propagating the noise points.

the progressive densification. Therefore, in a pyramid scheme, a possible strategy is to change λ according to the levels of the pyramid adaptively. In our method, linear interpolation from 0 to 0.5 is used when processing from the top level to the bottom level.

2.4. Bending energy for threshold adaption

As described above, the filter threshold must vary according to the different terrain features. To compensate for landform variations, the magnitude of the curvature must be quantitatively gauged. Furthermore, the relationship between the measurements and threshold compensations should be related to a parametric function. The bending energy of the TPS is used to measure the roughness, and it is simultaneously obtained using the calculation of the TPS (Elonen, 2005). The threshold that distinguishes the ground and non-ground points is compensated with the parameter *bend_gain*, which is related to the bending energy to account for complex terrain features. The *bend_gain* should be monotonously increasing for an increasingly rugged spot. Thus, the intuitive relationship between bending energy and *bend_gain* may be a linear interpolation after defining the upper bound (denoted as *max_bend_gain*). However, if extremely large values of the bending energy exist, a more robust approach is piecewise linear interpolation, as shown in Fig. 1c.

Unfortunately, we also discovered that roughness is only a necessary condition against *bend_gain*, rather than a sufficient condition. More explicitly, a greater roughness, in some cases, should not result in a larger compensation, as explained in Fig. 4. In the procedure of iterative filtering, some ground points on sharp ridges may be labeled as unclassified because of the local minimum constraint. It is probable that the TPS interpolated surface is lower than the ground surface, so the proposed relationship between bending energy and *bend_gain* is tenable in this situation. However, in a valley scenario, the relationship is not valid. Although the interpolated TPS will generate a large bending energy in this situation, it may be higher than the actual ground when regularization is adopted, and no compensation should be assigned. In this case, a binary mask raster *bend_mask* is used to amend the situation. The

bend_mask is assigned with the simultaneous interpolation of each grid. If the interpolated grid elevation is larger than the average elevation of its 12 nearest neighbor points, *bend_mask* = 1 is used, which means the *bend_gain* is positive. Otherwise, *bend_mask* = 0 is used to deny the compensation for the latter scenario. Therefore, the final compensation for the threshold is the product of *bend_gain* and *bend_mask* as in Eq. (1). We have found that this strategy is efficient for dealing with sharp ridges and large scarps.

3. Experimental results and analysis

To compare the proposed ASF filter with previous algorithms, benchmark datasets, provided by ISPRS Commission III, Working Group III (http://www.itc.nl/isprswgIII-3/filtertest/) (Vosselman, 2003), were employed to test our algorithm. Fifteen reference samples from seven sites are compiled with a binary indicator as either bare-earth or object points using semi-automatic filtering and subsequent manual editing. First, the evaluation paradigm used in the work by Sithole and Vosselman (2004) is adopted for quantitative and qualitative evaluations of performance, and explicit comparisons between ten other methods prove the high performance of the proposed method. In addition, we discuss and explain two of the characteristics of the ASF in detail. Furthermore, we analyze the effects of the three novel aspects using experimental validations and detailed discussions.

3.1. Performance evaluation and comparison

Three accuracy metrics were proposed by Sithole and Vosselman (2004) for the quantitative analysis of the filter performance, including Type I (*T.I*) error, Type II (*T.II*) error and Total error (*T.E.*), which represent the proportion of bare-earth points being misclassified as objects, failure to rejecting object points and the entire set of erroneous points, respectively (Sithole and Vosselman, 2004). Furthermore, Cohen's kappa coefficient (κ) (Congalton, 1991), which is a statistical measure of the inter-ratio agreement and is believed to be a more robust measurement than a simple percentage, has been widely used in previous works (Silván-Cárdenas
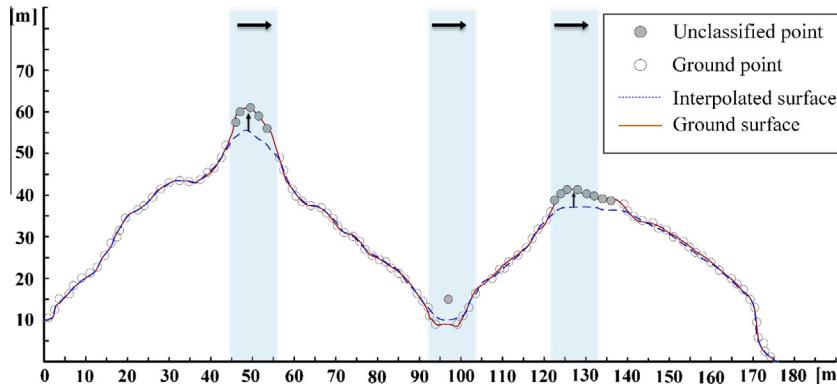
Fig. 4. Correction of the false bending compensation in case of the valley area. The upward arrows indicates the effects of the bending energy, which compensates for the filtering threshold to correctly label the unclassified points on the sharp ridges as ground points.

and Wang, 2006; Chen et al., 2013; Pingel et al., 2013). We also include a comparison of κ with other works in the present study. The report by Vosselman (2003) provided the calculations of *T.I*, *T.II* and *T.E.*, and the calculation of κ is presented intuitively and mathematically in the study of Congalton (1991). For completeness, we summarize these works in Table 1, and the four metrics are based on the inputs in the cross-matrix only (Sithole and Vosselman, 2004) as denoted by the shaded values in Table 1.

Three parameters are specifically tuned for the ASF, including the *max_window* for the window size of top level in the data pyramid, the threshold *t* as mentioned in Eq. (1) and the *max_bend_gain* for the upper bound of the bending energy compensation. To achieve the optimal results, we exhaust all the combinations from the configuration space of the three parameters in the ASF. The configuration space is defined using the three tuned parameters, where *max_window* $\in [5, 40]$ at the interval of 5 m, $t \in [0.1, 0.5]$ at the interval of 0.1 m and *max_bend_gain* $\in [0.1, 1]$ at the interval of 0.1 m. The range for each parameter was determined by the authors' practical experiences. For this configuration space, the results for a single reference sample consist of 400 tests. For each sample, the parameter set that achieves the minimum *T.E.* was

chosen as the optimum, as shown in Table 2. Furthermore, the configuration that achieves the best average *T.E.* among all the 15 samples was selected as the most stable parameter set.

Table 2 shows the ASF accuracy results tested against the 15 benchmark reference samples. With respect to the optimized results, the overall average total error and κ coefficient are 2.85% and 90.29%, respectively, and the median accuracies improved to 2.52% and 92.04% because the median is statistically more robust to gross error. In addition, for a single parameter set, an average *T.E.* of 3.67% and κ of 87.01% were achieved. In terms of the individual sample accuracy, samp11 has the largest total error, and samp53 has the worst κ statistic values. For the former sample, nearly all of the filters produce poor results due to the complex configuration of steep slopes, buildings and low vegetation. In addition, the achieved accuracy is already among the best. For the latter sample, the abnormally small κ is the result of the very low number of object points in that sample, which causes $Pr(e)$ to approach $Pr(a)$, with values of 0.92 and 0.9726, respectively, as denoted in Table 1, thus creating the poorest κ coefficient. However, the achieved result on samp53 is on par with the best (Pingel et al., 2013) and represents a significant improvement in

Table 1

Calculation equations of *T.I*, *T.II*, *T.E.* and κ for the quantitative evaluations of the ALS filtering algorithms. The four metrics are based on the four inputs (*a*, *b*, *c*, *d*) in the cross-matrix only where *a* and *d* are the numbers for correctly identified bare-earth and object points, respectively, and the other two parameters are the numbers of misclassified bare-earth and object points, respectively.

		Filtered			
		Bare-earth	Object		
Reference	Bare-earth	*a*	*b*	*a+b*	*f=(a+b)/e*
Reference	Object	*c*	*d*	*c+d*	*g=(c+d)/e*
		a+c	*b+d*	*e=a+b+c+d*	
		h=(a+c)/e	*i=(b+d)/e*		
T.I		*b/(a+b)*			
T.II		*c/(c+d)*			
T.E.		*(b+c)/e*			
κ	*Pr(a)*	*(a+d)/e*			
κ	*Pr(e)*	*f×h+g×i*			
κ	κ	*(Pr(a)-Pr(e))/(1-Pr(e))*			

Table 2
Performance evaluations of the ASF on *T.I*, *T.II*, *T.E.* and κ for the reference datasets provided by ISPRS and the corresponding parameters. For the single parameter set results, the parameters were selected as *max_window* = 30 m, *t* = 0.3 m, and *max_bend_gain* = 0.5 m.

Study site	Major terrain type	Parameter configurations			Optimized results				Results on single parameter set			
		max_window (m)	*t* (m)	*max_bend_gain* (m)	*T.I* (%)	*T.II* (%)	*T.E.* (%)	κ (%)	*T.I* (%)	*T.II* (%)	*T.E.* (%)	κ (%)
samp11	Steep slope	35	0.3	0.5	6.32	10.98	8.31	82.97	6.41	11.07	8.40	82.78
samp12	Flat	20	0.3	0.2	1.75	3.45	2.58	94.83	1.15	4.91	2.99	94.02
samp21	Bridge	30	0.3	0.1	0.36	3.03	0.95	97.23	0.03	8.56	1.92	94.26
samp22	Common	30	0.4	0.3	1.67	6.69	3.23	92.04	1.67	7.54	3.50	91.76
samp23	Complex	35	0.4	0.1	4.73	4.07	4.42	91.14	4.55	4.97	4.75	90.47
samp24	Ramp and vegetation	20	0.3	0.4	2.19	8.07	3.80	90.39	2.10	9.48	4.12	89.52
samp31	Negative blunders	30	0.2	0.1	0.50	1.37	0.90	98.19	0.06	7.00	3.26	93.41
samp41	Discontinuousness	25	0.4	0.2	2.95	8.86	5.91	88.18	3.00	9.52	6.27	87.47
samp42	Railway	35	0.4	0.3	0.39	0.87	0.73	98.25	0.47	1.52	1.21	97.10
samp51	Slope and low vegetation	30	0.2	0.2	0.53	7.46	2.04	93.90	0.12	12.30	2.78	91.49
samp52	Steep slope	20	0.2	1	1.07	14.90	2.52	86.24	2.25	11.09	3.18	83.69
samp53	Break lines	5	0.4	1	1.63	38.75	2.74	66.43	5.40	11.95	5.67	53.06
samp54	Village	15	0.2	0.2	2.33	2.36	2.35	95.28	1.51	3.74	2.71	94.57
samp61	Embankments	5	0.5	0.4	0.27	16.75	0.84	86.76	2.19	8.62	2.41	71.08
samp71	Bridge	30	0.4	0.4	0.93	5.93	1.50	92.59	0.75	10.56	1.86	90.54
Mean					1.84	8.90	2.85	90.29	2.11	8.19	3.67	87.01
Median					1.63	6.69	2.52	92.04	1.67	8.62	3.18	90.54
Maximum					6.32	38.75	8.31	98.25	6.41	12.30	8.40	97.10
Minimum					0.27	0.87	0.73	66.43	0.03	1.52	1.21	53.06

comparison with previous interpolation-based filters with the TPS (Chen et al., 2013).

To conclude the discussion of the performance results shown in Table 2, we will demonstrate two characteristics of the proposed ASF algorithm. First, a tendency to *T.II* error was discovered for both of the optimized results and those of a single configuration. *T.II* exceed *T.I* errors in almost all the samples, and the average *T.II* errors are approximately four times larger, as shown in Table 2. In fact, this tendency is also shared by previous methods based on the strategy of ground point densification (Axelsson, 2000; Mongus and Žalik, 2012; Chen et al., 2013; Zhang and Lin, 2013). The reasons for this inclination may be twofold. The first reason is that the numbers of object points in these datasets are generally less than the ground points, and sometimes the object points comprise only 30% or even approximately 3% (samp53 and samp61) of all the points. Thus, only a few object points being misclassified as bare-earth will result in a large *T.II* error. Second, a more explicit reason originates from the strategy of ground point densification. As previously stated, the ground points will be predominantly contaminated with objects in the iterative processing. Thus, the intermediate ground surface may have higher spikes or protuberances

above the ground. In return, the ground points are more likely to be correctly classified, thereby resulting in fewer *T.I* errors around the noise sources. However, the inclination to *T.II* errors may not be a flaw for the filter strategy, taking into consideration that *T.II* errors can be more easily handled by human editing than *T.I* errors (Sithole and Vosselman, 2004; Zhang and Lin, 2013).

The other characteristic is that the ASF method is not only superior in performance but also stable and insensitive to parameter tuning under varying terrain features and complex situations. These advantages are confirmed by the results from a single configuration that produces a T.E. of less than 5% in most of samples, and even the worst case still produces a low value of 8.4%, as shown in Table 2. Furthermore, we performed another experiment with varying configurations to confirm the robustness of the ASF. The configuration space in this experiment is *max_window* = 30 m, $t \in [0.1, 0.5]$ and *max_bend_gain* $\in [0.1, 0.5]$, which includes 25 configurations for a sample, and 375 results are obtained in total. The distribution of *T.E.* for all the results is shown in the histogram in Fig. 5. We find that most of the *T.E.* values are concentrated between 1% and 6%, which comprise approximately 75% of all the samples. In addition, 359 out of 375 tests produced results of *T.E.*
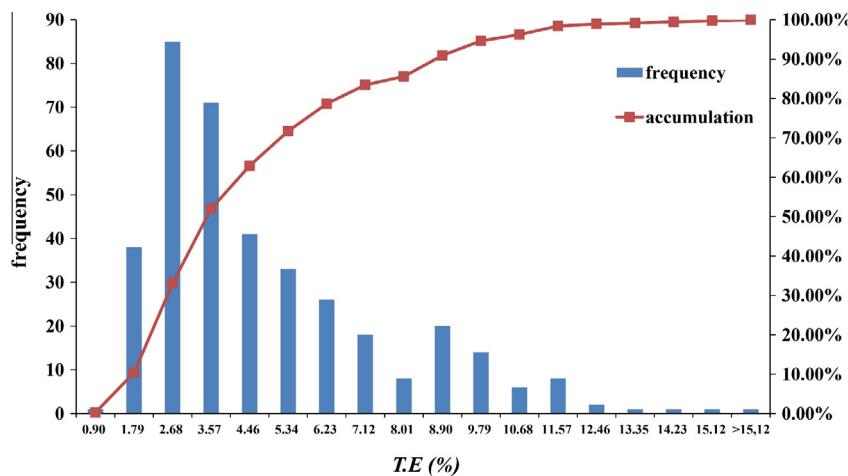

Fig. 5. Distribution of *T.E.* from tests on 15 reference samples with 25 parameter configurations (*max_window* = 30 m, $t \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ and *max_bend_gain* $\in \{0.1, 0.2, 0.3, 0.4, 0.5\}$). The blue bar shows the numbers of *T.E.* in a certain range, and the red line represents the accumulated percentage of the distribution. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
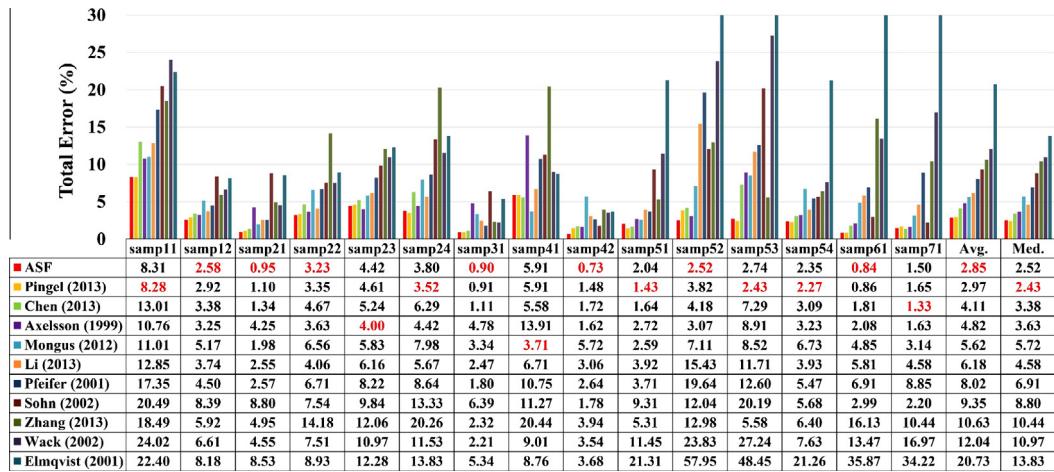
	samp11	samp12	samp21	samp22	samp23	samp24	samp31	samp41	samp42	samp51	samp52	samp53	samp54	samp61	samp71	Avg.	Med.
■ ASF	8.31	2.58	0.95	3.23	4.42	3.80	0.90	5.91	0.73	2.04	2.52	2.74	2.35	0.84	1.50	2.85	2.52
■ Pingel (2013)	8.28	2.92	1.10	3.35	4.61	3.52	0.91	5.91	1.48	1.43	3.82	2.43	2.27	0.86	1.65	2.97	2.43
■ Chen (2013)	13.01	3.38	1.34	4.67	5.24	6.29	1.11	5.58	1.72	1.64	4.18	7.29	3.09	1.81	1.33	4.11	3.38
■ Axelsson (1999)	10.76	3.25	4.25	3.63	4.00	4.42	4.78	13.91	1.62	2.72	3.07	8.91	3.23	2.08	1.63	4.82	3.63
■ Mongus (2012)	11.01	5.17	1.98	6.56	5.83	7.98	3.34	3.71	5.72	2.59	7.11	8.52	6.73	4.85	3.14	5.62	5.72
■ Li (2013)	12.85	3.74	2.55	4.06	6.16	5.67	2.47	6.71	3.06	3.92	15.43	11.71	3.93	5.81	4.58	6.18	4.58
■ Pfeifer (2001)	17.35	4.50	2.57	6.71	8.22	8.64	1.80	10.75	2.64	3.71	19.64	12.60	5.47	6.91	8.85	8.02	6.91
■ Sohn (2002)	20.49	8.39	8.80	7.54	9.84	13.33	6.39	11.27	1.78	9.31	12.04	20.19	5.68	2.99	2.20	9.35	8.80
■ Zhang (2013)	18.49	5.92	4.95	14.18	12.06	20.26	2.32	20.44	3.94	5.31	12.98	5.58	6.40	16.13	10.44	10.63	10.44
■ Wack (2002)	24.02	6.61	4.55	7.51	10.97	11.53	2.21	9.01	3.54	11.45	23.83	27.24	7.63	13.47	16.97	12.04	10.97
■ Elmqvist (2001)	22.40	8.18	8.53	8.93	12.28	13.83	5.34	8.76	3.68	21.31	57.95	48.45	21.26	35.87	34.22	20.73	13.83

Fig. 6. Comparison of *T.E.* for each reference sample with 10 previous works from 1999 to 2013. The ASF results are marked by the red bar, and the colored numbers in the following table denote the best results across all the methods. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
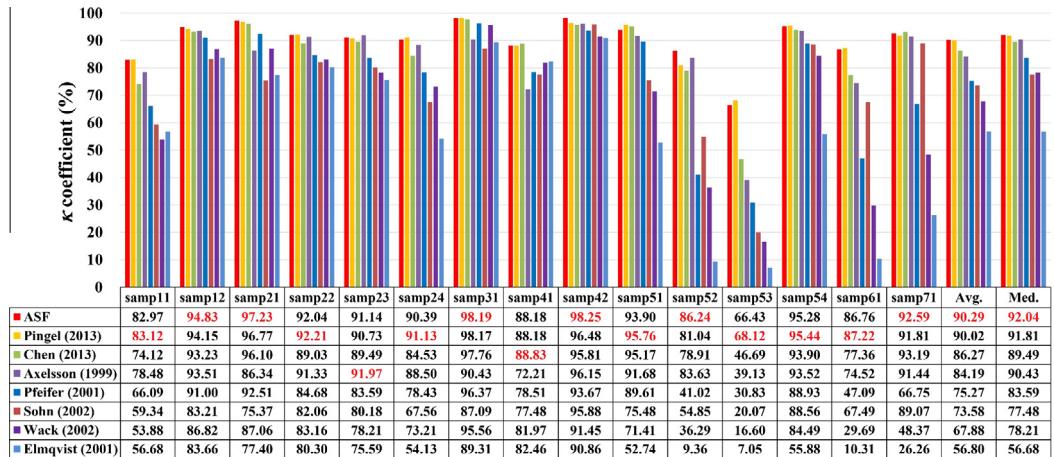

	samp11	samp12	samp21	samp22	samp23	samp24	samp31	samp41	samp42	samp51	samp52	samp53	samp54	samp61	samp71	Avg.	Med.
■ ASF	82.97	94.83	97.23	92.04	91.14	90.39	98.19	88.18	98.25	93.90	86.24	66.43	95.28	86.76	92.59	90.29	92.04
■ Pingel (2013)	83.12	94.15	96.77	92.21	90.73	91.13	98.17	88.18	96.48	95.76	81.04	68.12	95.44	87.22	91.81	90.02	91.81
■ Chen (2013)	74.12	93.23	96.10	89.03	89.49	84.53	97.76	88.83	95.81	95.17	78.91	46.69	93.90	77.36	93.19	86.27	89.49
■ Axelsson (1999)	78.48	93.51	86.34	91.33	91.97	88.50	90.43	72.21	96.15	91.68	83.63	39.13	93.52	74.52	91.44	84.19	90.43
■ Pfeifer (2001)	66.09	91.00	92.51	84.68	83.59	78.43	96.37	78.51	93.67	89.61	41.02	30.83	88.93	47.09	66.75	75.27	83.59
■ Sohn (2002)	59.34	83.21	75.37	82.06	80.18	67.56	87.09	77.48	95.88	75.48	54.85	20.07	88.56	67.49	89.07	73.58	77.48
■ Wack (2002)	53.88	86.82	87.06	83.16	78.21	73.21	95.56	81.97	91.45	71.41	36.29	16.60	84.49	29.69	48.37	67.88	78.21
■ Elmqvist (2001)	56.68	83.66	77.40	80.30	75.59	54.13	89.31	82.46	90.86	52.74	9.36	7.05	55.88	10.31	26.26	56.80	56.68

Fig. 7. Comparison of κ coefficient for each reference sample with 7 previous works from 1999 to 2013. The ASF results are marked by the red bar, and the colored numbers in the following table denote the best results across all the methods. Because κ was not provided in the works of (Mongus and Žalik, 2012; Li, 2013; Zhang and Lin, 2013), they are omitted here. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

values of less than 10%, which confirms that ASF can provide stable performance over a wide range of reasonable parameters. The robustness in the performance lies in the enhancements of the implementation details embedded in the entire workflow of the ASF. First, the pyramid scheme amended by a small step factor divides the original point clouds into more levels and results in the elevation differences between each level being smaller than that of the previous methods (Mongus and Žalik, 2012), which establishes a basis for handling noise in the filtering procedure. Second, the regularization applied in the TPS results in the interpolated ground surface being resilient to ground points with mixed objects. Finally, the bending energy is embedded into the filtering stage, which makes the proposed method adaptable to various difficult scenarios.

Turning the focus to a comparison with the previous filters, ten algorithms developed between 1999 and 2013 were used for the performance evaluations in terms of *T.E.* and the κ coefficient, as shown in Figs. 6 and 7. Briefly, the ASF achieves the best performance in seven and six of fifteen samples for *T.E.* and κ, respectively, and is the best in terms of the average values. Furthermore, for the remaining samples, the accuracy achieved is stable and among the best. When compared to the direct predeces-

sors of the ASF based on TPS interpolation (Mongus and Žalik, 2012; Chen et al., 2013), ASF achieves better results in almost all the cases and provides approximately 30% and 50% performance improvements in terms of the average *T.E.* Because fundamental strategies are commonly shared among these filters, accuracy enhancements must be attributed to the enhanced implementation details of the ASF. With regards to the comparison between the two filters based on PTD (Axelsson, 2000; Zhang and Lin, 2013), the ASF performance exceeds the performance of both of them. Although the surface represented by TIN is able to handle point density variations, the local ground surface is only relevant with three vertices and is expressed by a triangle that is simply a plane and is quite sensitive to noise. As a result, the filter threshold and strategy must be meticulously crafted. Otherwise, filters based on PTD may not yield the desired performance. In contrast, ASF exploits regularization embedded in the TPS interpolation to handle errors and is thus more robust. The improved morphological-based filter by Pingel et al. (2013) achieved comparable results compared to ours in almost all the samples with optimized parameters. However, with a fixed parameter set, the average *T.E.* decreases to 4.4%, which is approximately 20% inferior to that achieved by the ASF (3.67%). Furthermore, their method consists of four parameters,
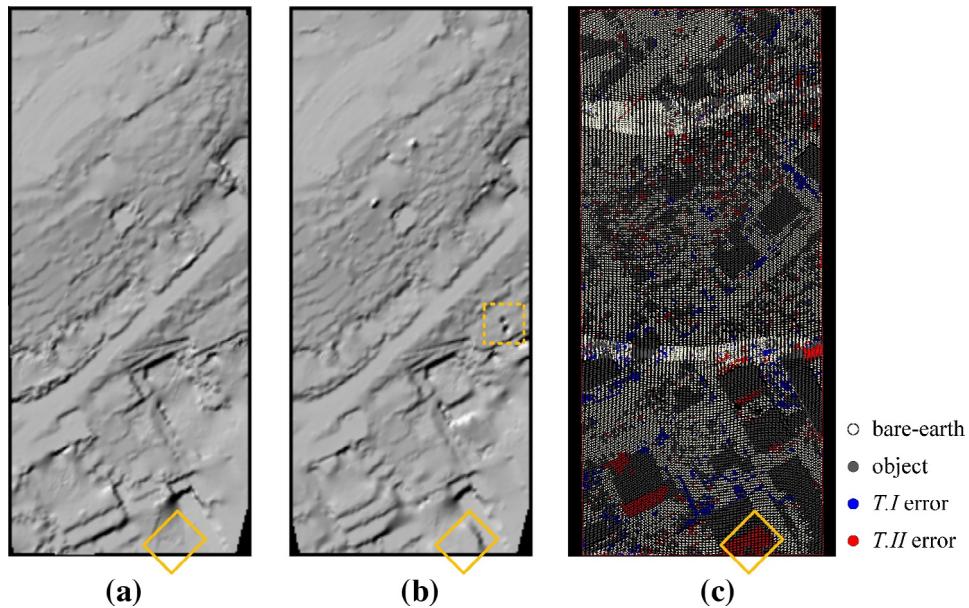
Fig. 8. Filtering results for samp11. (a) The reference DEM, (b) the filtered DEM, (c) the distribution of *T.I* and *T.II* errors. The two small holes indicated by the dashed rectangle are created by incorrect classification of low outliers in post-processing. The areas enclosed in the rectangles show the consequences of misclassification of low objects on the steep slope.
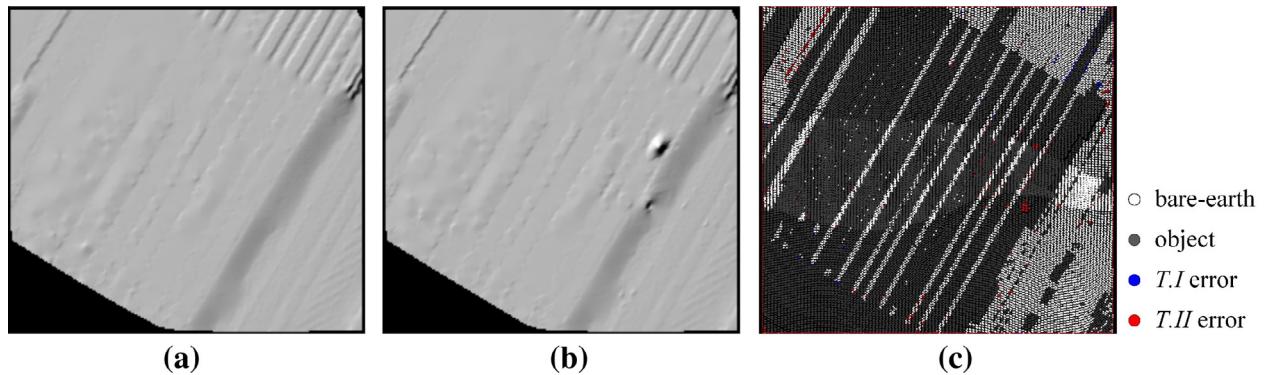

Fig. 9. Filtering results for samp42. (a) The reference DEM, (b) the filtered DEM, (c) the distribution of *T.I* and *T.II* errors. The overall accuracy is quite good except for the small protuberance because of several misclassified points on the roof of the platform that may be wrongly filtered because of a lack of nearby ground point.

which are tuned at a more fine-grained level and over a larger scope. In fact, the sensitivity to parameters is the major drawback for morphological-based filters when compared to the interpolated-based methods (Mongus and Žalik, 2012).

To evaluate qualitatively the performance of the ASF for various scenarios, we provide the filtering results for four samples as shown in Figs. 8–11. The four samples represent miscellaneous difficulties from urban to rural areas. First, samp11 is a complex mixture of buildings and vegetation on a steep slope. Second, samp42 is a railway station, where a low number of bare-earth points exist. Third, samp51 consists of data gaps and low vegetation on a slope. Lastly, samp71 has a bridge. The visualizations of the filtered DEMs exhibit nearly the same appearances with the reference DEMs in all the samples, and the error points are sparse in general. In fact, all the difficult scenarios have already been sufficiently addressed. However, we will explore further details for some recognizable differences between the filtered DEM and the reference DEM due to the complexity of the scenarios. The small holes in the middle of Fig. 8b as denoted by the dashed rectangle are the consequences of two misclassified low outlier points. The low outliers were actually detected in the preprocessing by the radius removal filter, but because

the elevations are close to the ground, they are incorrectly labeled as bare-earth in the post-processing. Furthermore, clamped *T.II* errors are found in samp11 due to the configuration of the low objects and the steep slopes. The rough area in samp42 along the slope, as shown in Fig. 10b, is due to the impacts of misclassified very low objects, most of which are only less than 0.5 m above the slope ground.

3.2. Experiments analysis

To analyze further the effects of the three contributions employed in this study, we performed three comparative experiments using the leave-one-out method with respect to the step factor, regularization effects and bending energy effects.

3.2.1. Effects of the step factor

To characterize the effects of the step factor when building the data pyramid rather than using a constant of two, we evaluated four samples, which contain buildings, slopes and vegetation, using varying step factors ($s \in [1.2, 2.2]$ at the interval of 0.05). The other parameters are the same as those in Table 2 and held fixed. The three types of error for the entire results are plotted in Fig. 12.
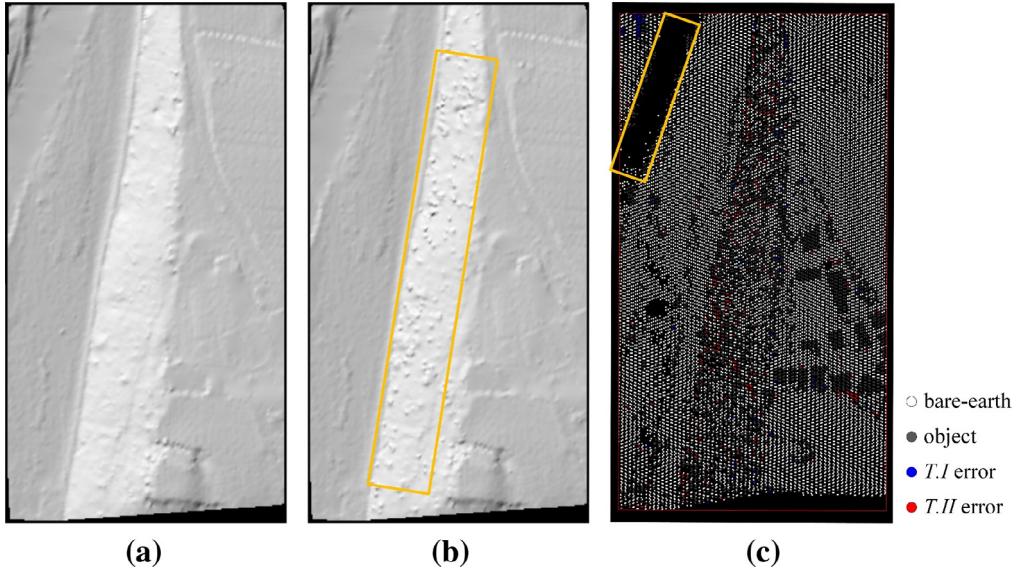
Fig. 10. Filtering results for samp51. (a) The reference DEM, (b) the filtered DEM, (c) the distribution of *T.I* and *T.II* errors. The roughness denoted by the rectangle in (b) along the slope is created by scattered *T.II* errors for the very low objects, and the data gap in (c) is handled well.
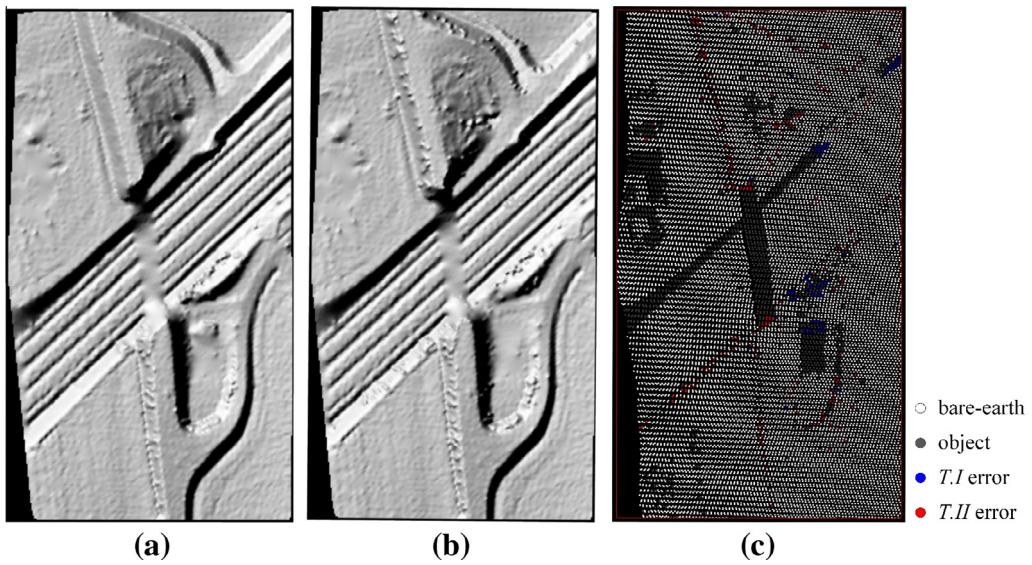

Fig. 11. Filtering results for samp71. (a) The reference DEM, (b) the filtered DEM, (c) the distribution of *T.I* and *T.II* errors. The bridge is correctly removed.

The relationship between the three error metrics and *s* is consistent in all samples: the *T.II* errors decrease moderately at the cost of a significant increase in the *T.I* errors, which thus results in an increasing total error. In addition, *T.II* errors exhibit no further noticeable improvements after *s* increases to 1.5.

The reason for the significant reduction of *T.I* errors with slowly increased window size (small *s*) can be explained by exploring the procedure of constructing the data pyramid. When building the data pyramid, points with minimum elevation are assigned to a previous level and then removed from the points that are responsible for the next level. With an inconspicuous difference between the adjacent pyramid levels, the elevation differences between the two levels are also subtle and thus accept more ground points. Accordingly, the TPS surface will gradually approach the final ground with more bare-earth points and fewer *T.I* errors. More bare-earth points will absolutely increase the risk of *T.II* error. However, because the errors are effectively handled by ASF, they are controlled to an acceptable level.

3.2.2. Effects of regularization

The regularization parameter λ, as described in Eq. (2), controls the balance between the fitness of the data and the smoothness of the surface. As discussed above, the merits of TPS with regularization for surmounting the occurrence of noise points in the iterative procedure are the reason for its excellence in ALS filtering. However, the loss of ground details accompanies the resilience to errors. We will illustrate both the advantages and disadvantages of regularization with experimental validations.

To confirm the advantages of the procedure, we used four samples to create the filter results, with and without regularization, whereas all other parameters remained the same. Fig. 13 presents the *T.II* errors overlaid onto the TIN surface generated from the filtered bare-earth points. In the top row for each sample, although there are some errors around the large building, these errors will not be magnified in the next iteration when regularization is applied. However, when regularization is deliberately turned off as shown in the bottom row of Fig. 13, even some points on the
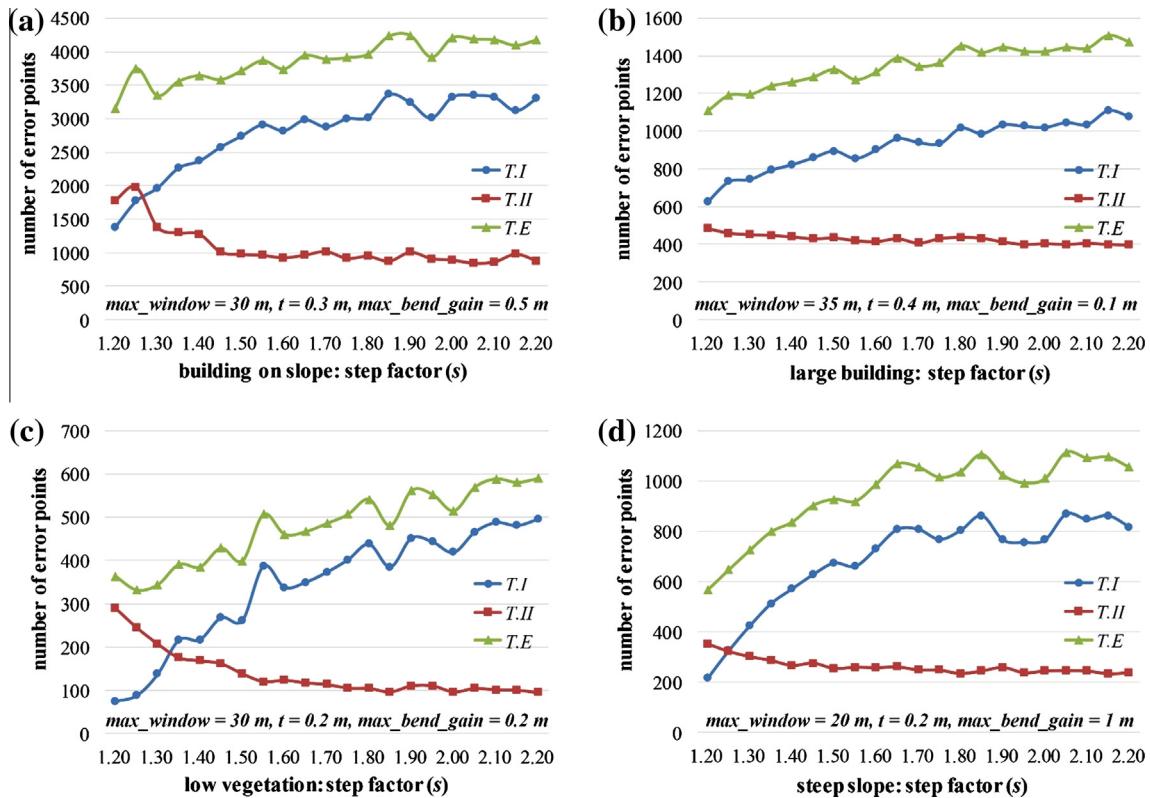
Fig. 12. Effects of the step factor in progressively building the data pyramid for the four samples on different scenarios: (a) samp11 for mixed buildings and vegetation on slope, (b) samp23 for typical building area, (c) samp51 for low vegetation and (d) samp52 for steep and terraced slopes. Conclusions can be drawn from various terrain types that *T.I* errors will greatly increase with a larger scale factor, and *T.II* errors will decrease mildly, which jointly results in an increasing of *T.E.*
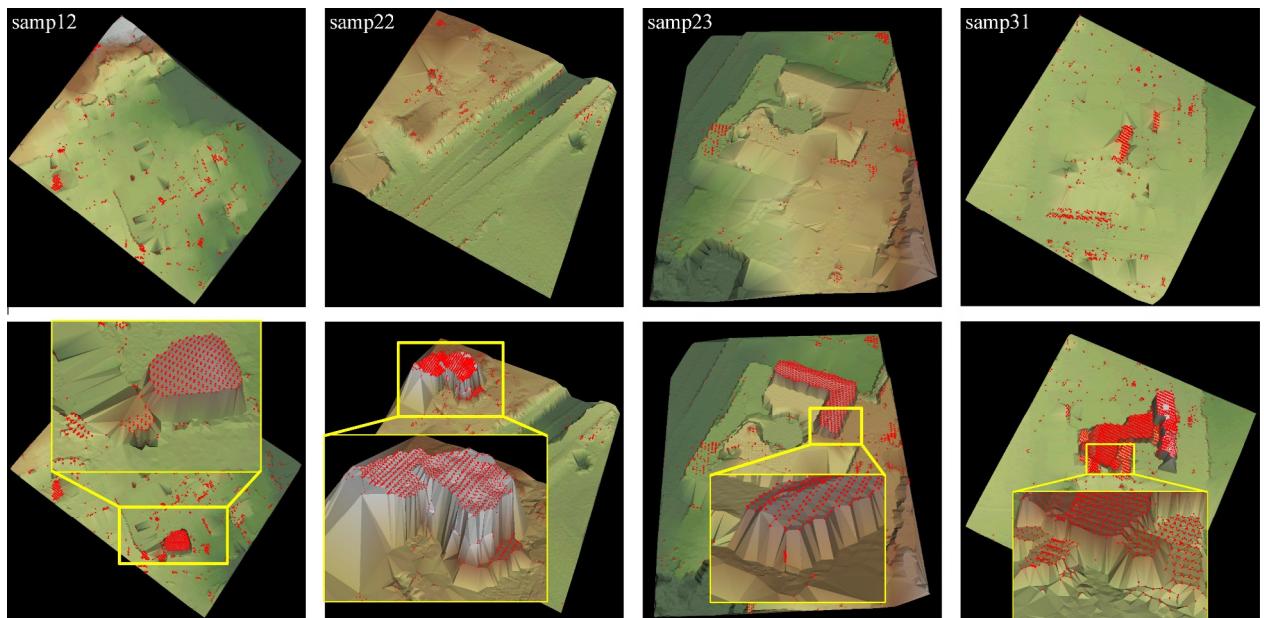

Fig. 13. Effects of regularization employed in TPS interpolation in four samples. The top row for each sample represents the T.II errors (red points) overlaid on the TIN surface with regularization ($\lambda = 0.5$), and the bottom row represents the T.II errors overlaid on TIN surface without regularization ($\lambda = 0$). The misclassified points consist of some low objects or points on the facade. Without regularization, the T.II errors will propagate through these points and climb up to the roof. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

rooftop will be misclassified. As explained in the magnified view for the building, some low points are attached to a small object and building facades, and they are first misclassified in the iterative procedure. Because no regularization is used in the interpolation, the surface must pass exactly through the points on the façade

or low objects. In the following iterations, the misclassified points clinging to the building act as the stairs and allow higher points on the building to be labeled as ground. When using proper regularization, a large smoothing penalty will occur at the spot of the misclassified noises because the elevation of an object is commonly
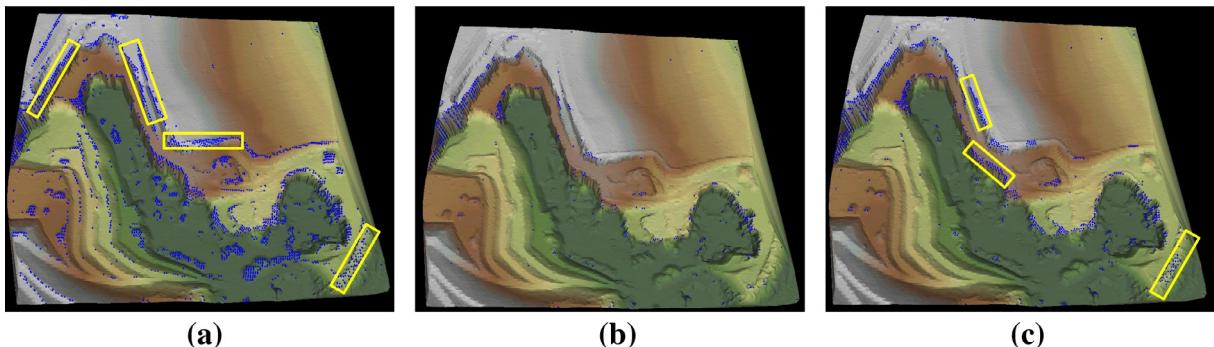
Fig. 14. *T.I* errors overlaid on samp53 with different parameter sets. (a) Filtering results with $\lambda = 0.5$, *max_bend_gain* = 0 m; (b) Filtering results $\lambda = 0.5$, *max_bend_gain* = 1.0 m; (c) Filtering results with $\lambda = 1.5$, *max_bend_gain* = 1.0 m. All the other parameters are the same with the optimized configurations as shown in Table 2. More *T.I* (blue points) errors are observed on the breaklines area denoted by rectangles in (a) and (c) due to lack of bending energy compensation and over-regularization, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

higher than its neighbor ground points (Sithole, 2001). Thus, the regularized surface will be lower than the elevation of the noise points as explained in Fig. 3c, and in this way, avoids the climbing-up effects. The regularization is essentially important for urban areas, where buildings may have low attachments, such as balconies and points on the façade, or where they will be affected by nearby low objects (cars or fences).

However, one drawback of regularization as described in the slope area in Fig. 3c is the loss of ground detail in the interpolation. An over-regularized surface will grant unnecessarily high weights on smoothness and result in poor data fit. Although smoothness is sometimes a tenable assumption for terrain, it fails under in situations with breaklines, e.g., samp53, as shown in Fig. 14c. Even with the same optimized parameter set, considerably more *T.I* errors exist on the breaklines when over-regularized compared to Fig. 14b, and the *T.E.* decreases to 4.08% accordingly. In fact, we achieved a better result even without regularization in samp53 with a *T.E.* of 2.41%, which is slightly better than the result in Table 2. Thus, over-regularization is inappropriate, and, in our experience, the case of $\lambda = 0.5$ (at the bottom level) is a good balance between the data term and the smoothness term that will consistently achieve stable performance.

3.2.3. Effects of the bending energy

Bending energy is used to remedy the problem of a single threshold against the varying terrain features. Samp53 was used to verify the effects of the compensation using bending energy because it possesses both flat terrain and breaklines. Fig. 14 compares the results without (by setting *max_bend_gain* = 0) and with the proposed compensation. In Fig. 14a, almost all the breaklines will cause *T.I* errors with a constant threshold, and the errors all exist in the higher part of a discontinuous area, as stressed by the rectangular region. This drawback is shared by almost all the previous surface filters in this sample (Sithole and Vosselman, 2004). However, with bending energy compensation (*max_bend_gain* = 1.0 m), the problem is relieved because, in places where a discontinuity exists, the interpolated TPS surface predominantly results in a larger bending energy. As a result, a larger compensation of the filtering threshold is required to correctly accept points on the edges of breaklines as ground.

4. Conclusions

Filtering of ALS point clouds is generally an obligatory step before its use in other applications and remains an open topic to date (Zhang and Lin, 2013). The proposed ASF uses an improved coarse-to-fine pyramid scheme. The window sizes between two adjacent

levels are multiplied by a step factor of 1.2 rather than a uniform factor of 2. The method iteratively interpolates the TPS surface to the bottom level. In the interpolation procedure, regularization is exploited to handle noise points, and the bending energy is retrieved to represent the roughness of the local surface and then used to adaptively change the filter threshold under varying terrain scenarios. As a result, the ASF displayed improved performance over its direct predecessor (Evans and Hudak, 2007; Mongus and Žalik, 2012; Chen et al., 2013) by a factor of approximately 30% in terms of the *T.E.* and was the best in terms of powerful when tested against the ISPRS benchmark datasets. Furthermore, the ASF can create more robust and good results, even when using the same parameter set, and improves the overall accuracy of the filter process. Future works will be devoted to developing a more sophisticated strategy for the compensation of local terrain structure. Furthermore, the full-waveform LiDAR (Mallet and Bretar, 2009) provides a new opportunity to integrate information extracted from the waveform energy into the classification process and to improve the performance and robustness of the algorithms further.

References

Axelsson, P., 2000. DEM generation from laser scanner data using adaptive TIN models. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 33 (Part B4), 111–118.
Chen, C., Li, Y., Li, W., Dai, H., 2013. A multiresolution hierarchical classification algorithm for filtering airborne LiDAR data. ISPRS J. Photogr. Remote Sens. 82, 1–9.
Chen, Q., Gong, P., Baldocchi, D., Xie, G., 2007. Filtering airborne laser scanning data with morphological methods. Photogr. Eng. Remote Sens. 73 (2), 175–185.
Congalton, R.G., 1991. A review of assessing the accuracy of classifications of remotely sensed data. Rem. Sens. Environ. 37 (1), 35–46.
Elonen, J., 2005. Thin Plate Spline Editor – An Example Program in C++. <http://elonen.iki.fi/code/tpsdemo> (accessed 02.07.13).
Evans, J.S., Hudak, A.T., 2007. A multiscale curvature algorithm for classifying discrete return LiDAR in forested environments. IEEE Trans. Geosci. Remote Sens. 45 (4), 1029–1038.
Filin, S., 2002. Surface clustering from airborne laser scanning data. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 34 (Part 3A), 119–124.
Flood, M., 2001. LiDAR activities and research priorities in the commercial sector. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 34 (Part 3/W4), 3–8.
Haala, N., Kada, M., 2010. An update on automatic 3D building reconstruction. ISPRS J. Photogr. Remote Sens. 65 (6), 570–580.
Haugerud, R.A., Harding, D.J., 2001. Some algorithms for virtual deforestation (VDF) of LiDAR topographic survey data. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 34 (Part 3/W4), 211–218.

Hauglin, M., Dibdiakova, J., Gobakken, T., Næsset, E., 2013. Estimating single-tree branch biomass of Norway spruce by airborne laser scanning. ISPRS J. Photogr. Remote Sens. 79, 147–156.

Kraus, K., Pfeifer, N., 1998. Determination of terrain models in wooded areas with airborne laser scanner data. ISPRS J. Photogr. Remote Sens. 53 (4), 193–203.

Li, Y., 2013. Filtering airborne LiDAR data by improved morphological method based on multi-gradient analysis. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 40 (Part 5/W1), 191–194.

Liu, X., 2008. Airborne LiDAR for DEM generation: some critical issues. Prog. Phys. Geogr. 32 (1), 31–49.

Maguya, A.S., Junttila, V., Kauranne, T., 2013. Adaptive algorithm for large scale DTM interpolation from LiDAR data for forestry applications in steep forested terrain. ISPRS J. Photogr. Remote Sens. 85, 74–83.

Mallet, C., Bretar, F., 2009. Full-waveform topographic lidar: state-of-the-art. ISPRS J. Photogr. Remote Sens. 64 (1), 1–16.

Meng, X., Currit, N., Zhao, K., 2010. Ground filtering algorithms for airborne LiDAR data: a review of critical issues. Remote Sens. 2 (3), 833–860.

Mongus, D., Žalik, B., 2012. Parameter-free ground filtering of LiDAR data for automatic DTM generation. ISPRS J. Photogr. Remote Sens. 67, 1–12.

Morse, B.S., Yoo, T.S., Rheingans, P., Chen, D.T., Subramanian, K.R., 2005. Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions. In: Proc. 32nd International Conference on Computer Graphics and Interactive Technique (SIGGRAPH2005), ACM, Los Angeles, CA, pp. 78–87.

Muja, M., Lowe, D.G., 2009. Fast approximate nearest neighbors with automatic algorithm configuration. In: Proc. 2009 International Conference on Computer Vision Theory and Application (VISSAPP2009), INSTICC Press, Lisboa, Portugal, 5–8 February, pp. 331–340.

Pfeifer, N., Stadler, P., Briese, C., 2001. Derivation of digital terrain models in the SCOP++ environment. In: OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Terrain Models, Stockholm, Sweden, 1–3 March. 13 p. (on CDROM).

Pingel, T.J., Clarke, K.C., McBride, W.A., 2013. An improved simple morphological filter for the terrain classification of airborne LIDAR data. ISPRS J. Photogr. Remote Sens. 77, 21–30.

Rusu, R.B., Cousins, S., 2011. 3D is here: Point Cloud Library (pcl). In: Proc. 2011 International Conference on Robotics and Automation (ICRA 2011), IEEE, Shanghai, China, 9–13 May, pp. 1–4.

Silván-Cárdenas, J.L., Wang, L., 2006. A multi-resolution approach for filtering LiDAR altimetry data. ISPRS J. Photogr. Remote Sens. 61 (1), 11–22.

Sithole, G., 2001. Filtering of laser altimetry data using a slope adaptive filter. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 34 (Part 3/W4), 203–210.

Sithole, G., Vosselman, G., 2004. Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds. ISPRS J. Photogr. Remote Sens. 59 (1), 85–101.

Sithole, G., Vosselman, G., 2005. Filtering of airborne laser scanner data based on segmented point clouds. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 36 (Part 3), 66–71.

Szeliski, R., 2011. Computer Vision: Algorithms and Applications, first ed. Springer, London.

Terzopoulos, D., 1988. The computation of visible-surface representations. IEEE Trans. Pattern Anal. Mach. Intell. 10 (4), 417–438.

Véga, C., Durrieu, S., Morel, J., Allouis, T., 2012. A sequential iterative dual-filter for LiDAR terrain modeling optimized for complex forested environments. Comput. Geosci. 44, 31–41.

Vosselman, G., 2000. Slope based filtering of laser altimetry data. Int. Arch. Photogr., Remote Sens. Spatial Inf. Sci. 33 (Part B3/2), 935–942.

Vosselman, G., 2003. ISPRS Test on Extracting DEMs from Point Clouds: A Comparison of Existing Automatic Filters. <http://www.itc.nl/isprswgIII-3/filtertest/> (accessed 22.07.13).

Vosselman, G., Maas, H., 2010. Airborne and Terrestrial Laser Scanning. Whittles Publishing, Caithness, UK.

Wehr, A., Lohr, U., 1999. Airborne laser scanning-an introduction and overview. ISPRS J. Photogr. Remote Sens. 54 (2), 68–82.

Zhang, J., Lin, X., 2013. Filtering airborne LiDAR data by embedding smoothness-constrained segmentation in progressive TIN densification. ISPRS J. Photogr. Remote Sens. 81, 44–59.

Zhang, K., Chen, S.C., Whitman, D., Shyu, M.L., Yan, J., Zhang, C., 2003. A progressive morphological filter for removing nonground measurements from airborne LiDAR data. IEEE Trans. Geosci. Remote Sens. 41 (4), 872–882.