applied sciences

MDPI

Article

Iterative Morphological Filtering for DEM Generation: Improving Accuracy and Robustness in Complex Terrains

Shaobo Linghu [1,2], Wenlong Song [1,2,*], Yizhu Lu [1,2], Kaizheng Xiang [1,2], Hongjie Liu [1,2], Long Chen [1,2], Tianshi Feng [1,2], Rongjie Gui [1,2], Yao Zhao [1,2] and Haider Abbas [1,2]

[1] State Key Laboratory of Simulation and Regulation of Water Cycle in River Basin, China Institute of Water Resources and Hydropower Research, Beijing 100038, China
[2] Research Center on Flood & Drought Disaster Reduction of the Ministry of Water Resources, Beijing 100038, China
* Correspondence: songwl@iwhr.com

Abstract

Accurate terrain modeling from high-resolution digital surface models (DSM) is critical for geosciences, geology, geomorphology, earthquake studies, and applied geology. However, existing filtering methods such as progressive morphological filtering (PMF), cloth simulation filtering (CSF), and progressive TIN densification (TIN) often struggle with complex topography and urban structures, leading to either excessive ground loss or incomplete object removal. Furthermore, some of these algorithms are only specialized for point cloud data and are not optimized for grid data. To address these limitations, we propose an iterative morphological filtering (IMF) algorithm that introduces a binary surface edge-segmentation strategy. The method refines object–ground separation by combining iterative morphological operations with block-based graph-cut stitching, thus enhancing continuity and accuracy in challenging terrain. Validation on UAV-derived DSM over the Haihe Basin in China and the ISPRS Vaihingen dataset shows that IMF achieves notable accuracy improvements: the Vaihingen test areas yielded an average Type I error of 8.93%, Type II error of 3.09%, overall accuracy of 80.85%, and Kappa coefficient of 0.7524, while the Haihe Basin test areas achieved Type I and II errors of 2.22% and 1.87%, overall accuracy of 89.32%, and a Kappa coefficient of 0.8706. These results demonstrate that IMF outperforms conventional methods by reducing both Type I and Type II errors, producing terrains highly consistent with real conditions. This innovation provides a robust and scalable solution for digital elevation models (DEM) generation from gridded DSM, offering significant value for large-scale environmental monitoring and flood risk assessment.

Keywords: DEM generation; DSM; morphological filtering

check for updates

Academic Editors: Francesco Benedetto and Fabio Tosti

Received: 23 September 2025
Revised: 27 October 2025
Accepted: 30 October 2025
Published: 31 October 2025

Citation: Linghu, S.; Song, W.; Lu, Y.; Xiang, K.; Liu, H.; Chen, L.; Feng, T.; Gui, R.; Zhao, Y.; Abbas, H. Iterative Morphological Filtering for DEM Generation: Improving Accuracy and Robustness in Complex Terrains. *Appl. Sci.* **2025**, *15*, 11683. https://doi.org/10.3390/app152111683

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Digital elevation models (DEM) provide a mathematical representation of the Earth's surface topography, effectively capturing terrain variation in an image-based format [1,2]. DEM have been extensively applied in a wide range of engineering disciplines, offering critical support for construction planning, hydrological modeling, and spatial analysis [3–5]. In addition to these engineering applications, DEM are also widely used in geological sciences, where their high topographic resolution enables detailed analysis of landforms and supports the understanding and mitigation of natural hazards such as landslides, floods, and earthquakes research [6–9]. However, digital surface models (DSM) are most

commonly generated directly from satellite imagery or UAV-derived point clouds [10–12], incorporating both bare-earth topography and above-ground elements such as vegetation and built structures, typically represented as gridded raster data [13,14]. To obtain an accurate DEM, surface features must be effectively removed from the DSM [15–17].

Two primary strategies exist for filtering DSM to generate DEM. The first operates directly on DSM point cloud data, commonly used in UAV-based LiDAR applications [18,19]. The second focuses on pre-gridded DSM data [20,21], which has become more widely adopted due to its enhanced interpretability and computational efficiency. In practical applications, therefore, developing a simple, efficient, and robust algorithm tailored to gridded DSM data is of particular value.

Over the past decades, numerous DSM filtering algorithms have been developed for both point cloud and raster datasets. Traditional approaches often rely on manual editing, where experts identify and remove surface points. While precise, such methods are time-consuming and labor-intensive, limiting their scalability. With advances in automated image processing, a wide variety of filtering techniques have emerged, most of which are grounded in the assumption that surface objects exhibit abrupt elevation changes, whereas terrain varies more smoothly. For instance, Vosselman introduced a slope-based algorithm [22,23], which assumes that slopes between surface and ground points are steeper than those between ground points alone. Although computationally efficient, this method performs poorly in complex terrain due to its dependence on a fixed slope threshold. Similarly, Kilian proposed a morphological filtering approach [24], applying opening operations to detect surface objects by comparing height differences before and after morphological processing. However, its effectiveness strongly depends on the sliding window size: smaller windows fail to capture large objects, while larger ones risk removing natural terrain features. To address this limitation, Zhang introduced progressive morphological filtering [25], employing varying window sizes and weighted results. This strategy alleviates the window-size problem but still struggles with steep terrain, where over-removal remains a challenge. Axelsson's progressive TIN densification algorithm [26] further enhanced filtering accuracy by iteratively densifying a ground surface model from initial seed points. Nonetheless, its sensitivity to seed point selection and high computational costs hinder broader application. More recently, Zhang proposed the cloth simulation filter [27], which inverts the DSM and simulates a virtual cloth settling under gravity to delineate ground surfaces. This method is intuitive and requires fewer parameters but still demands terrain-specific tuning. Beyond traditional methods, Oshio introduced the pix2pix model [28], which applies deep learning to generate DEMs from DSMs. This approach demonstrates great potential; however, its performance may be influenced by the availability of large, high-quality training datasets and by its adaptability across different spatial scales.

In light of these challenges, this study proposes an iterative morphological filtering algorithm for DEM generation from gridded DSM, termed IMF. This method is built on two core assumptions: (1) surface features exhibit significant elevation differences compared to the terrain [29], and (2) these features form closed sets of elevated grid cells above the ground [30]. Leveraging morphological operations [31], IMF constructs a terrain surface and incrementally elevates it to segment the DSM into a sequence of binarized images. Boundaries extracted from these binary layers are then compared with differences between the original DSM and its eroded version to identify and isolate surface features. These identified features are subsequently set to null and interpolated to yield the final DEM.

The proposed IMF method addresses critical limitations of existing DSM filtering techniques by combining the conceptual simplicity of morphological operations with an iterative design that enhances adaptability to complex terrain. Unlike conventional

fixed-window or parameter-sensitive approaches, IMF provides a balance between computational efficiency and filtering accuracy. Its emphasis on closed-set feature detection makes it especially effective in urban and heterogeneous environments, while its iterative nature reduces over-removal in steep regions. Consequently, IMF not only advances the methodological toolkit for DEM generation but also offers significant practical relevance for large-scale, high-resolution applications where efficiency and robustness are equally crucial. In particular, the algorithm demonstrates strong potential in a wide range of geospatial and environmental applications, including topographic mapping, geological hazard assessment (e.g., landslides and fault zone analysis), watershed and flood modeling, infrastructure construction planning, land-use monitoring, and ecological change detection, where accurate terrain representation is fundamental to reliable analysis.

2. Methodology

Apart from simple denoising preprocessing (i.e., filtering to remove outlier noise caused by birds or anomalous reflections from objects [32]), the iterative morphological filtering (IMF) algorithm consists of six major modules: image blocking, morphological grayscale filtering, iterative binarization, connected component labeling, threshold filtering, and surface interpolation (see Figure 1). Among these, image blocking, morphological grayscale filtering, connected component labeling, and interpolation are standard image-processing operations, whereas iterative binarization and threshold filtering constitute the core innovations of the algorithm (see Figure 2).

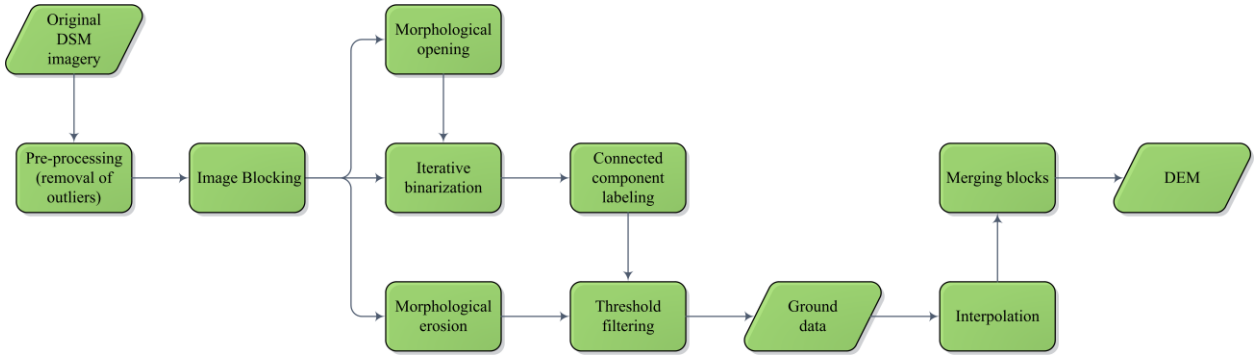


Figure 1. Flowchart of the IMF algorithm.

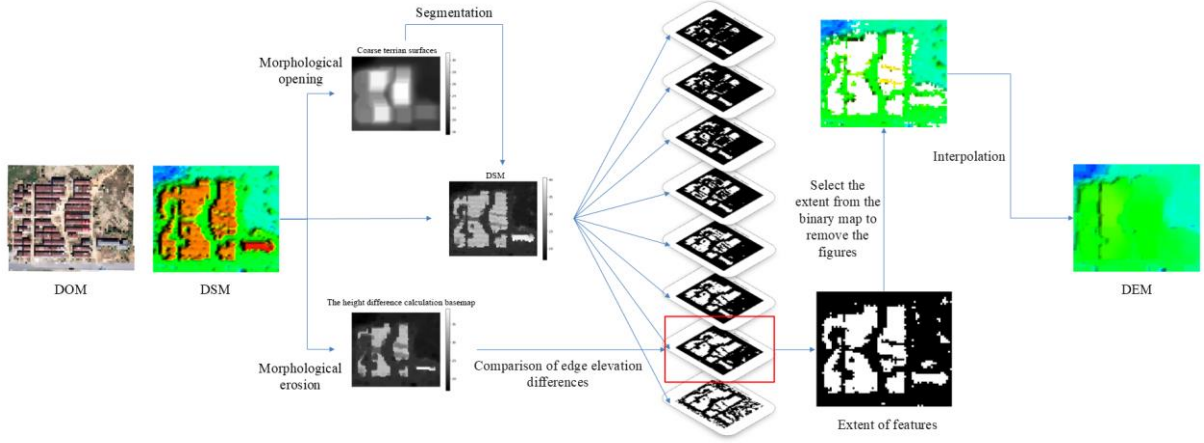


Figure 2. Illustration of the core process of the IMF algorithm. The changes in the image during processing are shown. DOM (digital orthophoto map) is an orthorectified reference image.

2.1. Image Blocking

When processing large remote sensing images with tens of millions or even billions of pixels, dividing the image into blocks improves memory efficiency, avoids memory overflow, and reduces computational complexity [33]. The block size is chosen to be larger than the maximum object size in the study area; specifically, the block dimensions in both horizontal and vertical directions exceed the largest extent of object contours. This ensures that each block fully contains the boundaries between objects and ground. To further avoid discontinuities across block boundaries, overlapping areas are defined between adjacent blocks (see Figure 3). The pixel values in these overlapping regions are then optimized using a graph-cut algorithm [34] to determine seamless stitching paths.

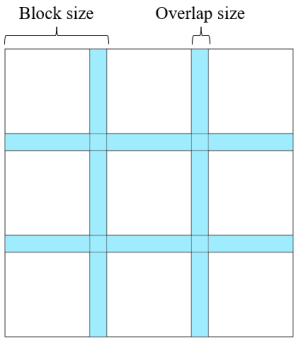


Figure 3. Diagram of block processing.

2.2. Morphological Grayscale Filtering

Morphological grayscale filtering [35], an extension of mathematical morphology, is applied to the digital surface model (DSM). In this study, morphological opening and erosion operations are used:

$$E(x,y) = \min_{(i,j)\in S}\{I(x+i,y+j)\}, \tag{1}$$

$$O(x,y) = \text{Dilation}(\text{Erosion}(I(x,y))), \tag{2}$$

where $I(x,y)$ is the original image, $E(x,y)$ is the erosion operation, $O(x,y)$ is the opening operation, and S is the structural element.

With a suitably large structuring element (larger than the maximum object radius), the opening operation produces a coarse digital elevation model (DEM). A subsequent erosion with a smaller structuring element (e.g., less than 7×7) transfers ground values to object-edge pixels, facilitating later comparison. The opening result serves as the threshold basemap for iterative binarization, while the erosion result provides the reference for height-difference calculations in subsequent steps.

2.3. Iterative Binarization

To distinguish ground from non-ground pixels, a binary threshold must be determined. However, in areas with strong topographic variability, a single threshold is insufficient. Therefore, an iterative binarization strategy is applied [36].

In this approach, the threshold basemap is incrementally raised by a fixed iteration step (typically 0.25–0.5 m), which is then used as the segmentation threshold for the DSM (see Figure 4). This produces a sequence of binary images, each representing potential object areas. Foreground regions in these images are provisionally marked as candidate objects for further refinement. A large iteration step may overlook low-elevation objects, while a small step may lead to redundant computation:

$$B_t(x,y) = \begin{cases} 1, & Z_t(x,y) \geq F_t(x,y) \\ 0, & Z_t(x,y) < F_t(x,y) \end{cases}, \tag{3}$$

$$F_{t+1}(x,y) = F_t(x,y) + \Delta F, \tag{4}$$

where $B_t(x,y)$ represents the binary image, $F_t(x,y)$ is the threshold basemap, $Z_t(x,y)$ is the original DSM, and ΔF is the iteration value.

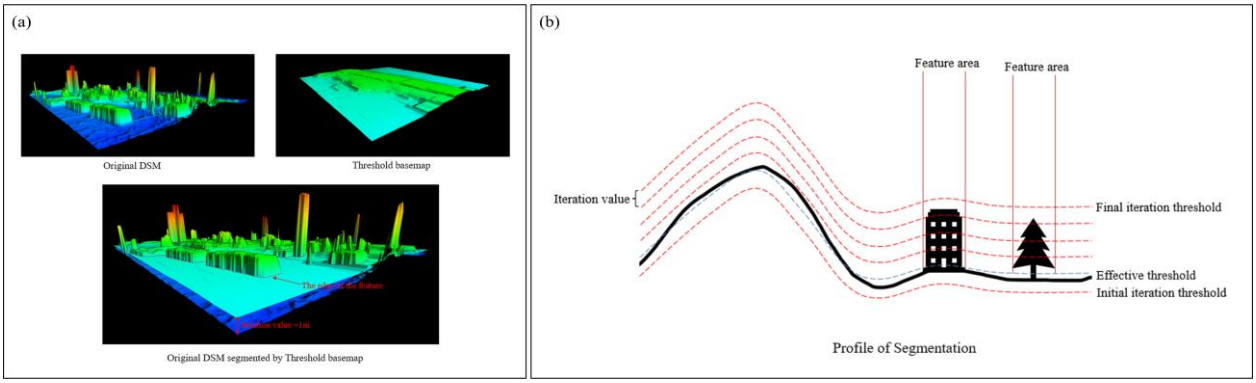


Figure 4. Diagram of iterative binarization. (**a**) Original DSM segmented by Threshold basemap when iteration value is 1 m. (**b**) Profile of Segmentation. When iteration threshold reaches the Effective threshold, the feature area is identified.

2.4. Connected Component Labeling

The binary images obtained from iterative binarization are subjected to connected component labeling to identify contiguous clusters of foreground pixels [37,38]. The algorithm scans each pixel and, if a foreground pixel is detected, assigns a label to it. Its four immediate neighbors (up, down, left, right) are then checked. If a labeled neighbor exists, the same label is assigned; otherwise, a new label is created. Each label represents a distinct object candidate. This study adopts the four-connectivity approach:

$$f_{4(x,y)} = \{f(x+1,y), f(x-1,y), f(x,y+1), f(x,y-1)\}, \tag{5}$$

where $f_{4(x,y)}$ represents the extracted connected area, while $f(x,y)$ is the binary image.

2.5. Threshold Filtering

From the labeled components, filtering is applied in two steps. First, a maximum pixel count threshold is enforced: any component exceeding the maximum expected object size in the study area is excluded to reduce computational load. Second, for the remaining components, edge pixels are identified by checking adjacency to background pixels. The average elevation of edge pixels is compared against the corresponding average from the height-difference basemap (derived from morphological erosion). If this difference exceeds a predefined threshold, the component is classified as an object cluster and removed by assigning its pixels as NoData.

2.6. Surface Interpolation

After object removal, interpolation is performed to fill the resulting voids. A smoothing filter is then applied to the interpolated surface to generate a refined DEM, representing the true ground surface without above-ground objects.

3. Experiments

3.1. Comparative Algorithms

DSM filtering algorithms were tested on DSM from various areas with different geographic conditions to verify the stability of the IMF algorithm. The results of several mainstream DSM filtering algorithms—namely, progressive morphological filtering (PMF) [25,39], cloth simulation filtering (CSF) [27], and progressive TIN filtering (TIN) [26,40]—were also obtained for quantitative comparison, illustrating the advanced performance of the IMF algorithm.

In order to reduce the poor performance of the results caused by different parameter settings, all the comparison algorithms set different parameters for processing, and finally select the best results for comparison. For some point clouds filtering algorithms, the method of converting the gridded data into point clouds for processing is adopted, and finally the results are presented in a grid.

3.2. Accuracy Metrics

The filtering results for the DSM data are evaluated using Type I error, Type II error, overall Accuracy, and the Kappa coefficient to assess the effectiveness of object removal [41]:

$$\text{Type I Error} = \frac{\text{FP}}{\text{FP} + \text{TN}}, \tag{6}$$

$$\text{Type II Error} = \frac{\text{FN}}{\text{FN} + \text{TP}}, \tag{7}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{8}$$

$$P_e = \frac{(\text{TP} + \text{FP}) \times (\text{TP} + \text{FN}) + (\text{TN} + \text{FN}) \times (\text{TN} + \text{FP})}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{9}$$

$$Kappa = \frac{P_o - P_e}{1 - P_e}, \tag{10}$$

where TP represents true positives, TN represents true negatives, FP represents false positives, FN represents false negatives, P_o represents observed agreement, which refers to the actual predictive accuracy of the model (accuracy), and P_e represents random agreement, which is the probability that the predicted results are randomly consistent with the true results.

In this study, Type I error represents the proportion of ground points incorrectly identified as objects, indicating cases where true terrain surfaces are mistakenly removed during filtering. Type II error denotes the proportion of object points incorrectly retained as ground, reflecting insufficient removal of above-ground features. Accuracy measures the overall proportion of correctly classified pixels, including both ground and non-ground areas. The Kappa coefficient further evaluates the agreement between the filtering results and the reference data by considering the possibility of random consistency, providing a more robust assessment of the classification reliability.

Based on the delineated object boundaries within the study area, the region was divided into ground (TN) and non-ground (TP) zones, serving as the reference ground truth. The difference between the algorithm-derived ground area and its intersection with the reference ground region represents the missed object area (FN), while the difference between the algorithm-derived non-ground area and its intersection with the reference non-ground region represents the over-detected object area (FP).

3.3. Datasets and Results Analysis

3.3.1. Dataset 1

Based on the ISPRS Vaihingen dataset (DSM with 25 cm resolution, generated from UAV point clouds), three test areas (see Figure 5) exhibiting typical terrain and object characteristics were selected to compare the object removal performance of IMF, CSF, PMF, and TIN algorithms. The test areas cover various complex scenes, including steep slopes with building clusters, elongated buildings with dense vegetation, and gentle terrain with isolated buildings and individual trees, to evaluate the algorithm's adaptability in heterogeneous environments.

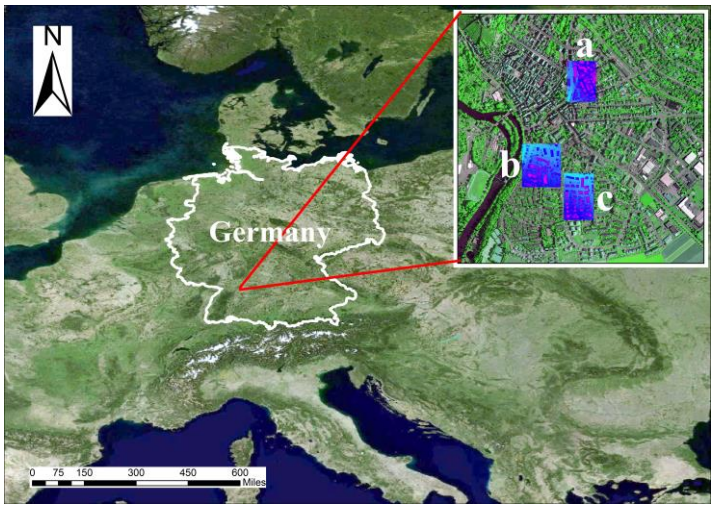


Figure 5. The geographical location of Dataset1, with optical images and DSM images in the top-right thumbnail both from the ISPRS Vaihingen dataset, displays the distribution of test areas a, b, and c.

Area a (Figure 6a) is mainly characterized by contiguous standalone buildings of varying sizes and a small amount of vegetation adjacent to the buildings, with significant slope variations. Area b (Figure 6b) primarily consists of elongated buildings and densely clustered vegetation, with very pronounced slope variations that test the algorithm's adaptability to densely vegetated areas and steep terrain. Area c (Figure 6c) has relatively gentle slopes, and its features are mostly standalone buildings and isolated vegetation, which are rather sparsely distributed.

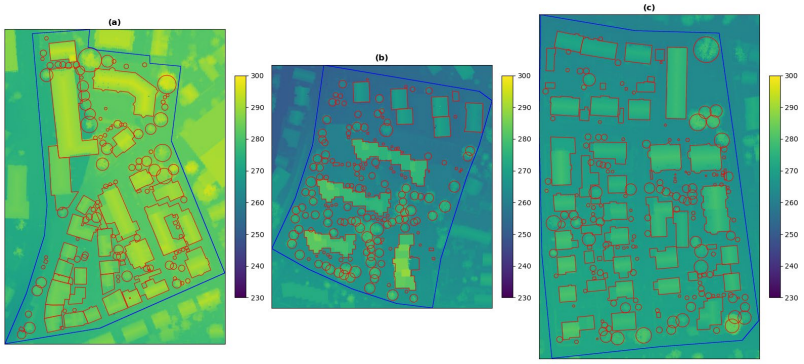


Figure 6. The test areas of Dataset 1 overlaid with the feature extent. The areas enclosed by the red line represent the features, while the areas enclosed by the blue line represent the actual accuracy verification calculation range. (**a**) Overview of Area a; (**b**) Overview of Area b; (**c**) Overview of Area c.

From Figure 7, it can be observed that in Area a (see Figure 7a), the IMF algorithm performs well, successfully separating buildings and vegetation in the steep slope area.

The surface features are clearly defined, effectively distinguishing terrain undulations and abrupt elevation changes in the features. The CSF algorithm in Area a exhibits a higher Type I error rate, classifying some undulating surfaces as features, and also fails to filter out the large building roofs on the right side, despite this large building not being included in the ground truth used for accuracy validation. The PMF algorithm misses some low vegetation and large building roofs on the right side and fails to differentiate buildings from steep slopes in areas with higher gradients. The TIN algorithm, when dealing with the large building group on the upper slope of Area a, incorrectly identifies large areas of the surface as features, resulting in the highest Type I error rate.

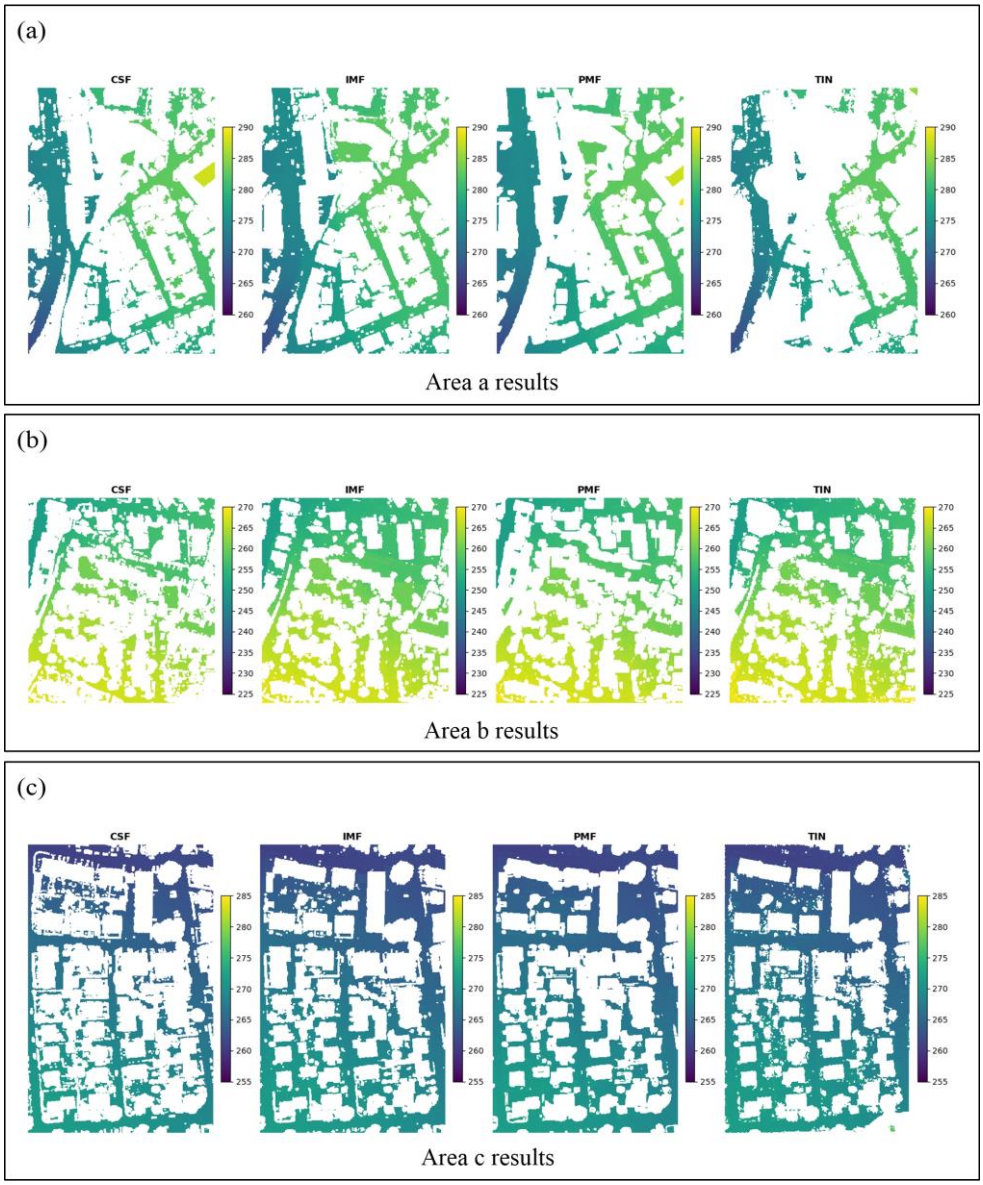


Figure 7. Comparison of ground object removal results for Dataset 1. (**a**) Area a results. (**b**) Area b results. (**c**) Area c results. White areas represent the ground objects identified by the algorithm.

In Area b (see Figure 7b), the IMF algorithm effectively removes most of the manually labeled features, even in steep areas, with a relatively low error rate. The CSF algorithm shows varying degrees of over-filtering in Area b, failing to distinguish small gaps between many features and removing a significant number of small terrain undulations, leading to a higher Type I error rate. The TIN algorithm filters out large areas of the surface around independent buildings in the upper right of Area b. However, the overall performance is

acceptable, with error rates remaining low. The PMF algorithm performs poorly in Area b, mistakenly classifying slopes as features in the upper slope area and incorrectly removing many gaps between features as features themselves. The Type I error rate exceeds 20%.

In Area c (see Figure 7c), all algorithms perform relatively well. The IMF algorithm excels in feature removal. The CSF algorithm, in Area c, performs well in most areas, with only a few narrow ground errors between features. The TIN algorithm benefits from the flat terrain, improving accuracy, but the high Type II error rate exposes its insensitivity to low vegetation. The PMF algorithm misses some smaller individual vegetation, with an overall average performance.

Seeing from Table 1 and Figure 8, in terms of Type I error rate, the IMF algorithm performs excellently in all three areas, with an average error rate of less than 10%. Except for Area c, the error rate is the lowest among all algorithms. The CSF algorithm has the highest average Type I error rate. For Type II error rate, the IMF algorithm also performs well, with an average error rate of around 3%, while the TIN algorithm has the highest average Type II error rate. In terms of accuracy, the IMF algorithm achieves around 80%, with the highest accuracy in both Area a and Area b. The Kappa coefficient for the IMF algorithm is consistently superior, indicating a high level of consistency between the IMF algorithm and the true feature classification results.

Table 1. Accuracy evaluation table of results for Dataset 1.

Test Area	Algorithm	Type I Error	Type II Error	Accuracy	Kappa Coefficient
a	CSF	19.20%	2.06%	71.20%	0.5762
	IMF	**10.99%**	3.50%	**80.72%**	**0.7106**
	PMF	18.65%	2.38%	71.65%	0.5807
	TIN	25.85%	**1.46%**	65.01%	0.4562
b	CSF	19.32%	**1.14%**	63.29%	0.5974
	IMF	**9.16%**	2.90%	**77.50%**	**0.7439**
	PMF	21.89%	1.45%	60.13%	0.5471
	TIN	11.14%	2.78%	73.98%	0.7085
c	CSF	14.36%	**1.30%**	72.21%	0.6893
	IMF	6.64%	2.88%	84.32%	**0.8027**
	PMF	8.67%	2.70%	80.55%	0.7668
	TIN	**4.85%**	6.12%	**87.00%**	0.7671

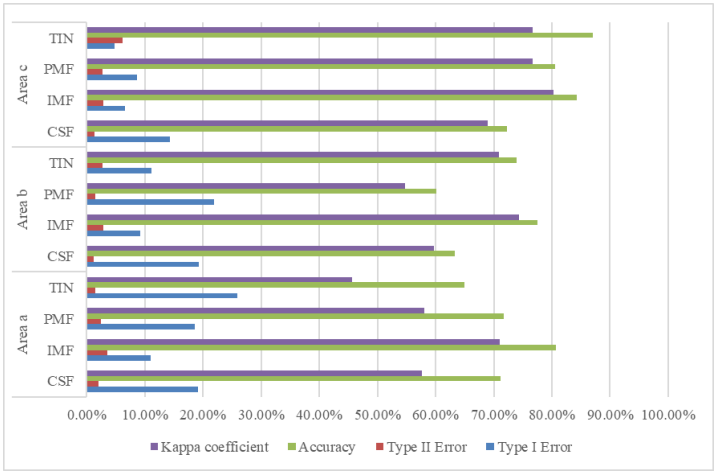


Figure 8. Accuracy evaluation chart of results for Dataset 1.

To more clearly and intuitively present the DEM results obtained by the IMF algorithm, profile plots for the three test areas of Dataset 1 processed by the IMF algorithm are shown in Figure 9.

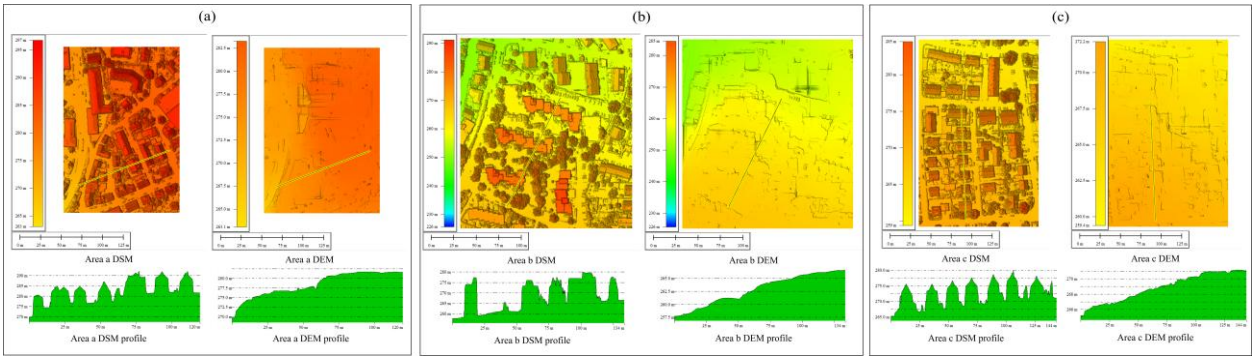


Figure 9. Dataset 1 profile processed by the IMF. (**a**) Topography and profile of Area a before and after IMF treatment. (**b**) Topography and profile of Area b before and after IMF treatment (**c**) Topography and profile of Area c before and after IMF treatment. The terrain profile corresponding to the yellow–black line segment in the topography image.

In Figure 9, the IMF algorithm has preserved the terrain well without disrupting the landscape, effectively maintaining the terrain's natural features.

3.3.2. Dataset 2

At the same time, based on DSM images with 10 cm resolution obtained from UAV point clouds in the Haihe Basin in China (see Figure 10), accuracy validation of the results processed by the IMF, CSF, PMF, and TIN algorithms was conducted through manual delineation of feature boundaries, in order to evaluate the adaptability of the IMF algorithm in complex field environments in China.

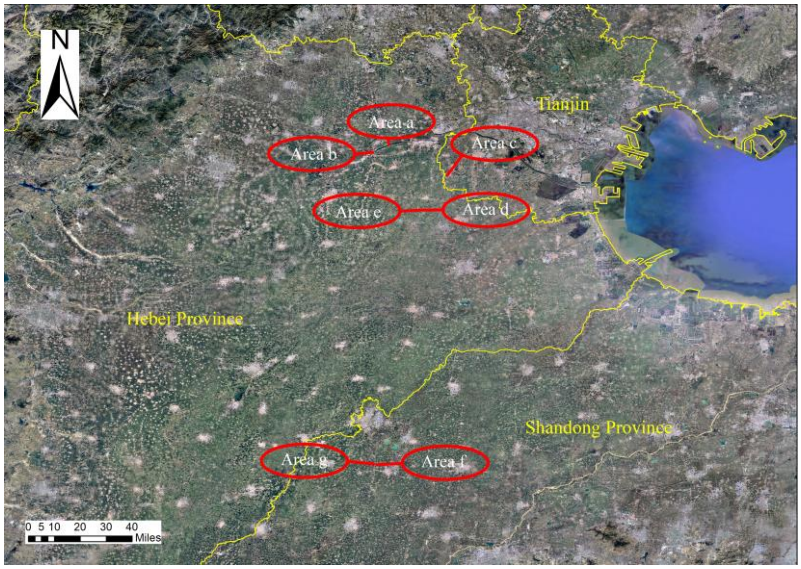


Figure 10. Geographical distribution map of the DSM test areas in Dataset 2. The seven test areas (Area a, b, c, d, e, f, g) are within the red circles, and the lines connecting to the red circles point to the sampling locations.

Figure 11 shows Dataset 2 and its test results, and Figure 12 shows the accuracy evaluation table and statistical chart of the test results. There are a total of seven test areas (see Figure 10(Area a)): (Area a) The image contains large factory buildings, scattered crops,

and dense forests; (Area b) the image includes large buildings, numerous trucks, and a bridge; (Area c) the image features small residential buildings, ponds, and vegetation along the shore; (Area d) the image shows abundant vegetation along riverbanks with a steep slope; (Area e) the image features a highway with a large number of shrubs along its side, with a steep slope; (Area f) the image depicts densely packed residential buildings, with some vegetation interspersed; (Area g) the image contains a large number of independent plants along a highway, with a steep slope.

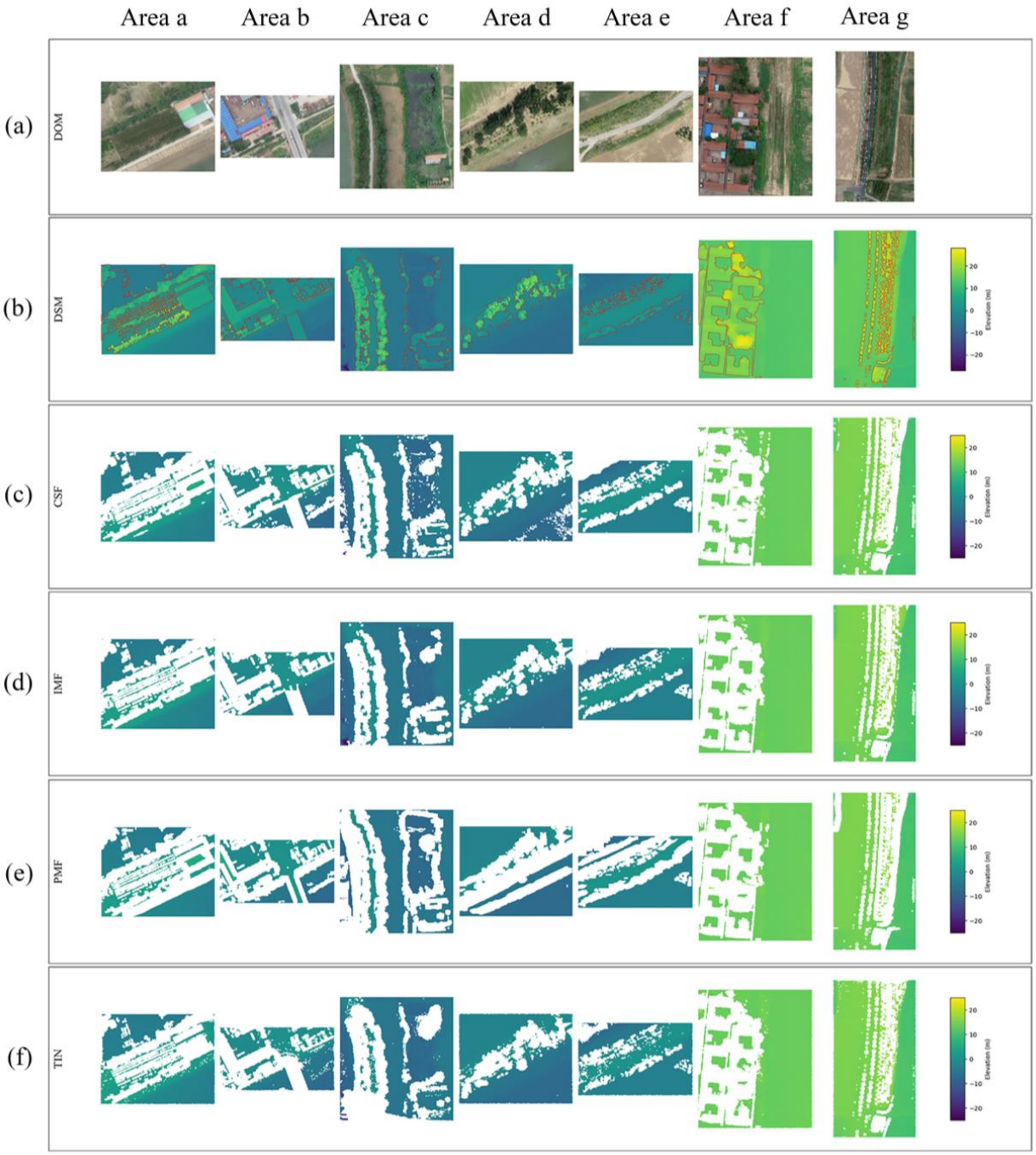
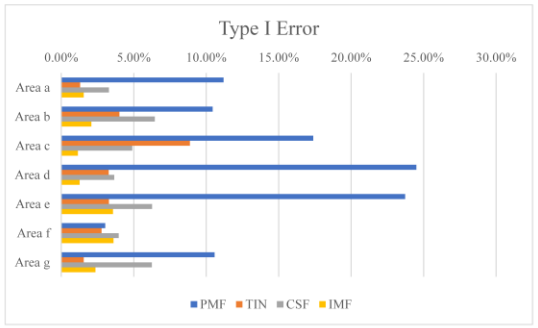


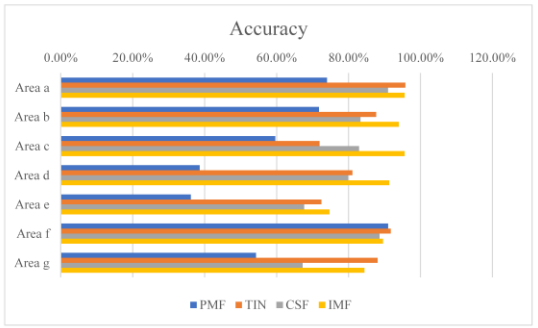
Figure 11. Dataset 2 and test result images. (**a**) Digital orthophoto map of the test areas. (**b**) The DSM with the overlaid feature boundaries, where the red line encloses the feature area. (**c**) The feature removal results of CSF. (**d**) The feature removal results of IMF. (**e**) The feature removal results of PMF. (**f**) The feature removal results of TIN. The white areas represent the feature boundaries identified by the algorithms.

(a)



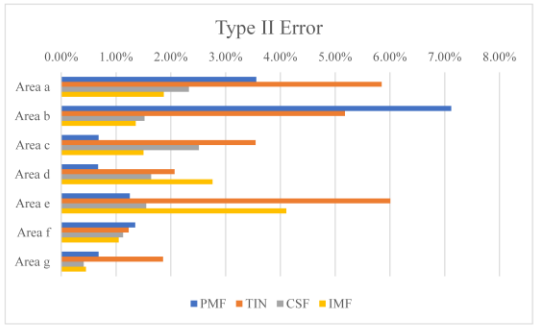
	PMF	TIN	CSF	IMF
Area a	11.19%	**1.31%**	3.29%	1.54%
Area b	10.45%	4.01%	6.45%	**2.08%**
Area c	17.38%	8.87%	4.90%	**1.15%**
Area d	24.50%	3.28%	3.65%	**1.27%**
Area e	23.72%	**3.29%**	6.26%	3.57%
Area f	3.03%	**2.78%**	3.97%	3.60%
Area g	10.58%	**1.54%**	6.25%	2.36%

(b)



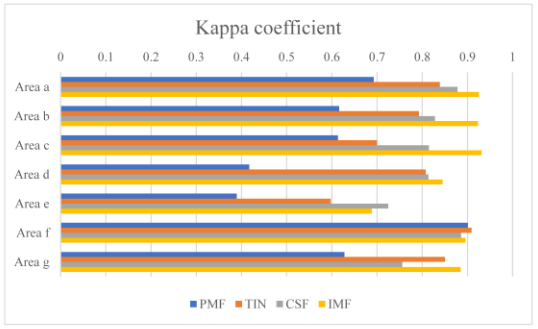
	PMF	TIN	CSF	IMF
Area a	74.01%	**95.77%**	90.96%	95.62%
Area b	71.78%	87.69%	83.31%	**93.97%**
Area c	59.61%	71.98%	82.95%	**95.58%**
Area d	38.65%	81.07%	79.86%	**91.32%**
Area e	36.13%	72.46%	67.69%	**74.71%**
Area f	91.03%	**91.73%**	88.64%	89.61%
Area g	54.29%	**88.08%**	67.25%	84.45%

(c)



	PMF	TIN	CSF	IMF
Area a	3.56%	5.85%	2.33%	**1.87%**
Area b	7.12%	5.18%	1.52%	**1.36%**
Area c	**0.68%**	3.55%	2.51%	1.50%
Area d	**0.67%**	2.07%	1.64%	2.76%
Area e	**1.25%**	6.01%	1.55%	4.11%
Area f	1.35%	1.23%	1.13%	**1.05%**
Area g	0.68%	1.86%	**0.41%**	0.45%

(d)



	PMF	TIN	CSF	IMF
Area a	0.6926	0.8388	0.8779	**0.9253**
Area b	0.616	0.7927	0.8279	**0.9234**
Area c	0.6133	0.6995	0.8146	**0.9315**
Area d	0.4171	0.8079	0.8135	**0.8452**
Area e	0.3896	0.5976	**0.7246**	0.6884
Area f	0.9008	**0.9092**	0.8856	0.8956
Area g	0.628	0.8503	0.7557	**0.8849**

Figure 12. Accuracy evaluation and statistical chart of each algorithm for Dataset 2. (**a**) Type I Error and their statistical table for each algorithm in Dataset 2. (**b**) Accuracy and their statistical table for each algorithm in Dataset 2. (**c**) Type II Error and their statistical table for each algorithm in Dataset 2. (**d**) Kappa coefficient and their statistical table for each algorithm in Dataset 2.

The performance of the PMF algorithm varies significantly across the seven test areas. Overall, its average accuracy is 60.79%, with an average Kappa coefficient of 0.6082. It performs best in Area f (dense residential buildings and interspersed vegetation), achieving an accuracy of 91.03% and a Kappa coefficient of 0.9008, indicating strong adaptability in complex urban environments. However, in Area d (steep riverbank vegetation) and Area e (steep shrubbery along the highway), the accuracy drops to 38.65% and 36.13%, respectively, with Type I error rates as high as 24.50% and 23.72%. These issues are likely due to the steep slopes and noise from water bodies in the images. Additionally, in Area c (small residential buildings and ponds), the Type I error rate (17.38%) is higher, which is likely due to the low elevation within the water body affecting morphological processing. In Area b, a bridge is also missed. Overall, the PMF algorithm is suitable for flat or moderately complex environments but performs poorly in steep or noisy areas.

The TIN algorithm has an average accuracy of 84.11% and an average Kappa coefficient of 0.7851, which is significantly better than the PMF algorithm. It performs particularly well in Area a (large factory buildings and forests) and Area f, with accuracies of 95.77% and 91.73%, respectively, showing robustness in large-scale feature segmentation. The Kappa coefficients for Area d and Area g (steep vegetation) increase to 0.8079 and 0.8503, indicating that TIN's triangular network modeling effectively mitigates the impact of terrain undulations. However, in Area e (highway steep slope shrubbery), the Kappa coefficient is only 0.5976, likely due to the small elevation differences between the vehicles and shrubs, leading to a higher Type II error rate (6.01%). TIN performs stably in complex terrains but has limited ability to distinguish small-scale features or targets with similar elevations.

The CSF algorithm has an average accuracy of 80.09%, which is close to that of TIN, but its Kappa coefficient (0.8143) is higher, indicating better classification consistency. The accuracy for Area a and Area b (large building areas) is 90.96% and 83.31%, respectively, showing CSF's strong ability to identify man-made features. However, in Area e and Area g (steep vegetation), the accuracy drops to 67.69% and 67.25%, mainly due to low Type II error rates (1.55%, 0.41%) but high Type I error rates (6.26%, 6.25%), reflecting a tendency to overly filter out non-feature targets. CSF performs moderately in balancing Type I and Type II errors and is suitable for areas with clearly defined feature boundaries and moderate to low slopes.

The IMF algorithm performs the best overall, with an average accuracy of 89.32% and a Kappa coefficient of 0.8706, significantly higher than the other algorithms. In Area a and c (factory buildings, small residential buildings, ponds, and shrubbery), the accuracy exceeds 95%, with Kappa coefficients above 0.92, demonstrating its high-precision ability to distinguish multiple feature types. In Area d (steep riverbank vegetation), the Kappa coefficient is 0.8452, indicating that IMF effectively reduces terrain interference through surface iteration. Only in Area e (highway steep slope) does it perform somewhat weaker (accuracy of 74.71%), but other algorithms also struggle in this area. Overall, IMF demonstrates exceptional robustness in complex environments and can effectively handle scenes with intertwined features and steep slopes.

4. Discussion

The experimental results from both datasets reveal several common limitations in existing filtering algorithms.

For instance, the cloth simulation filtering (CSF) algorithm encounters significant challenges in areas with steep terrain [42,43], where balancing fabric stiffness and flexibility is difficult. In Area a of Dataset 1 and Area a of Dataset 2, the roofs of large buildings were not completely removed due to insufficient stiffness, while higher stiffness values would have caused excessive filtering on sloped terrain.

Similarly, the progressive TIN filtering (TIN) algorithm exhibits poor adaptability on slopes. Elevated points on slope tops that exceed the preset height threshold from the triangular mesh are often incorrectly filtered out. Moreover, seed point selection is influenced by the distribution of surface features [44]. Dense vegetation or buildings can hinder the identification of suitable seed points [45], leading to unevenly distributed seeds and inaccurate ground triangulation. Increasing seed density can improve accuracy but significantly extends computation time. Consequently, in our tests, TIN was the most time-consuming method and performed suboptimally in Area a of Dataset 1 and Area c of Dataset 2.

The progressive morphological filtering (PMF) algorithm also tends to alter the original terrain morphology due to its opening and closing operations, which can excessively remove ground points even in regions with minor elevation variations [46]. Its main limita-

tion is that, despite progressively increasing the window size and height threshold, it cannot adapt well to complex surface structures [47]. For example, in Area b of Dataset 2, only the edges of a bridge were removed, while part of the bridge surface remained unfiltered.

In contrast, the proposed iterative morphological filtering (IMF) algorithm treats surface features as objects—sets of pixels—rather than evaluating individual pixels in isolation. By assessing the height difference at the boundaries of these objects, IMF reduces misclassification on steep slopes and preserves true terrain continuity, representing a substantial advancement over conventional filtering techniques.

4.1. Treatment of Steep Slopes

As illustrated in Figure 13, the elevation difference between the original surface and the threshold basemap at the slope top exceeds 1 m. Assuming a threshold of 1 m, conventional pixel-based methods would classify the slope top as a feature. IMF, however, uses an edge-based regional judgment strategy: height differences are calculated using a small structural element for erosion, evaluating the slope as an integrated object. This allows the slope top to be correctly retained as ground, improving slope preservation accuracy in complex terrains. However, this strategy requires DSMs with relatively clear feature boundaries.

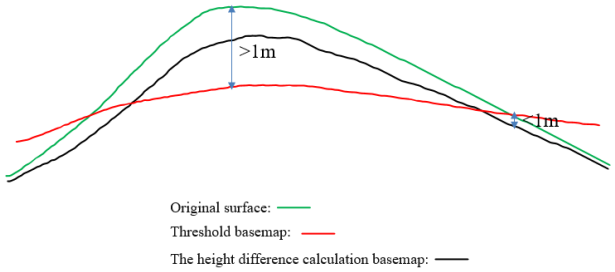


Figure 13. The judgment of steep slopes by the IMF algorithm.

4.2. Quantitative Performance and Error Trade-Off

In DSM filtering, Type I errors (misclassifying ground as objects) and Type II errors (misclassifying objects as ground) are inevitable and often inversely related. Most algorithms reduce one error type at the cost of increasing the other. IMF achieves a balanced trade-off, keeping both errors at relatively low levels. As a result, IMF consistently achieves higher overall accuracy and Kappa coefficients compared with other algorithms in both datasets.

The Kappa coefficient is particularly useful because it accounts for the possibility of agreement occurring by chance, providing a more reliable measure than overall accuracy alone [48]. For example, in regions where objects occupy only a small fraction of the area, a high overall accuracy might still correspond to poor filtering performance. The Kappa coefficient avoids this bias, making it a valuable complement to traditional accuracy metrics.

4.3. Limitations and Scope of Application

IMF performs best on high-resolution DSMs with spatial resolutions finer than 10 m, where feature boundaries are clearly defined and edge-based calculations remain reliable. As DSM resolution decreases, feature edges become blurred, negatively affecting IMF performance.

Regarding its applicability, the IMF algorithm is particularly well-suited for high-resolution DSM datasets derived from UAVs and large-scale photogrammetry, where object boundaries are relatively well-defined. It demonstrates strong performance in urban environments with sparse vegetation and maintains good adaptability in mountainous terrains,

areas where traditional algorithms often struggle due to steep slopes or irregular surface structures. In extremely flat regions dominated by low vegetation, however, additional refinement strategies, such as integration with machine learning classifiers, may be necessary to achieve optimal results.

5. Conclusions

This study presents an iterative morphological filtering (IMF) algorithm for generating high-precision DEMs from gridded DSM data. IMF integrates morphological filtering, iterative binarization, connected domain labeling, and edge-based height difference calculation, treating surface features as objects rather than individual pixels. This object-oriented approach enables more accurate discrimination between ground and non-ground elements, particularly in steep or complex terrains.

Experimental results on the ISPRS Vaihingen dataset and UAV DSM data from the Haihe Basin, China, demonstrate that IMF outperforms conventional algorithms such as PMF, TIN, and CSF. It effectively reduces both Type I and Type II errors, achieving higher overall accuracy and Kappa coefficients (80.85% and 0.7524 for the Vaihingen dataset; 89.32% and 0.8706 for the Haihe Basin dataset), highlighting its robustness across diverse landscapes.

IMF is especially suitable for high-resolution DSMs (better than 10 m) with clearly defined feature boundaries. It excels in areas with steep slopes, pronounced terrain undulations, and complex surface structures, addressing common limitations of traditional methods such as over-filtering or under-filtering. Its computational efficiency also makes it practical for large-scale topographic mapping, high-precision DEM generation, and geoscientific applications including urban planning, hydrological modeling and landslide analysis.

Despite these strengths, IMF may still face challenges in extremely complex or low-resolution terrains. Future work will focus on optimizing parameter selection, integrating deep learning techniques for improved adaptability, and extending the method to diverse remote sensing datasets with varying sensors and resolutions to enhance its generalizability.

Author Contributions: S.L.: Methodology, software, writing—original draft, validation; W.S.: conceptualization, supervision, funding acquisition; Y.L.: project administration, supervision; K.X.: writing—review and editing, formal analysis; H.L.: resources, data curation; L.C.: investigation, resources; T.F.: investigation, resources; R.G.: investigation, resources; Y.Z.: writing—review and editing; H.A.: writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by Hunan Water Conservancy Science and Technology under grant XSKJ2023059-04; in part by the Three Gorges Follow-up Work under grant JZ110161A0012025; and in part by Special Project on Basic Scientific Research Funds of China Institute of Water Resources and Hydropower Research under grant JZ110145B0062025.

Data Availability Statement: Dataset2 for this article are available at https://doi.org/10.5281/zenodo.15734386 (accessed on 29 October 2025); Iterative Morphological Filtering Algorithm. Contact: linghushaobo@edu.iwhr.com; Language: Python3.9; Requirements: GDAL3.8.4 (Geospatial Data Abstraction Library), SciPy1.13.1; Freely available at https://github.com/saoerbo/Iterative-Morphological-Filtering.git (accessed on 29 October 2025). Note: The IMF algorithm was entirely developed by our team in Python. It can be executed on any Python platform. Before use, users should adjust certain parameters according to the DSM resolution, as described in the README section.

Acknowledgments: The authors would like to thank the International Society for Photogrammetry and Remote Sensing for providing the 2D Semantic Labeling—Vaihingen data. The Vaihingen data (Dataset1) used are available at https://www.isprs.org/resources/datasets/benchmarks/UrbanSemLab/2d-sem-label-vaihingen.aspx (accessed on 29 October 2025).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Guth, P.L.; Van Niekerk, A.; Grohmann, C.H.; Muller, J.-P.; Hawker, L.; Florinsky, I.V.; Gesch, D.; Reuter, H.I.; Herrera-Cruz, V.; Riazanoff, S.; et al. Digital Elevation Models: Terminology and Definitions. *Remote Sens.* **2021**, *13*, 3581. [CrossRef]
2. Balasubramanian, A. *Digital Elevation Model (DEM) in GIS*; University of Mysore: Mysore, India, 2017.
3. Sestras, P.; Badea, G.; Badea, A.C.; Salagean, T.; Roșca, S.; Kader, S.; Remondino, F. Land surveying with UAV photogrammetry and LiDAR for optimal building planning. *Autom. Constr.* **2025**, *173*, 106092. [CrossRef]
4. Zhu, H.; Chen, Y. A study of the effect of DEM spatial resolution on flood simulation in distributed hydrological modeling. *Remote Sens.* **2024**, *16*, 3105. [CrossRef]
5. Sohl, M.A.; Mahmood, S.A. Low-cost UAV in photogrammetric engineering and remote sensing: Georeferencing, DEM accuracy, and geospatial analysis. *J. Geovisualization Spat. Anal.* **2024**, *8*, 14. [CrossRef]
6. Cirillo, D.; Tangari, A.C.; Scarciglia, F.; Lavecchia, G.; Brozzetti, F. UAV-PPK Photogrammetry, GIS, and Soil Analysis to Estimate Long-Term Slip Rates on Active Faults in a Seismic Gap of Northern Calabria (Southern Italy). *Remote Sens.* **2025**, *17*, 3366. [CrossRef]
7. Tempa, K.; Peljor, K.; Wangdi, S.; Ghalley, R.; Jamtsho, K.; Ghalley, S.; Pradhan, P. UAV technique to localize landslide susceptibility and mitigation proposal: A case of Rinchending Goenpa landslide in Bhutan. *Nat. Hazards Res.* **2021**, *1*, 171–186. [CrossRef]
8. Tabunshchik, V.; Gorbunov, R.; Gorbunova, T.; Pham, C.N.; Klyuchkina, A. Identification of river basins within northwestern slope of Crimean Mountains using various digital elevation models (ASTER GDEM, ALOS World 3D, Copernicus DEM, and SRTM DEM). *Front. Earth Sci.* **2023**, *11*, 1218823. [CrossRef]
9. Hidayat, E.; Muslim, D.; Wibowo, D.A.; Puswanto, E.; Aribowo, S.; Mulyono, A.; Sudrajat, Y. Appraisal of active tectonics in Karangsambung Amphitheater: Insights from DEM-derived geomorphic indices and geological data. In *IOP Conference Series: Earth and Environmental Science*; IOP Publishing: Bristol, UK, 2024; p. 012092.
10. Ressl, C.; Brockmann, H.; Mandlburger, G.; Pfeifer, N. Dense Image Matching vs. Airborne Laser Scanning—Comparison of two methods for deriving terrain models. *Photogramm. Fernerkund. Geoinf.* **2016**, *2016*, 57–73. [CrossRef]
11. Zheng, Z.; Wan, Y.; Zhang, Y.; Hu, Z.; Wei, D.; Yao, Y.; Zhu, C.; Yang, K.; Xiao, R. Digital surface model generation from high-resolution satellite stereos based on hybrid feature fusion network. *Photogramm. Rec.* **2024**, *39*, 36–66. [CrossRef]
12. Serati, G.; Sedaghat, A.; Mohammadi, N.; Li, J. Digital surface model generation from high-resolution satellite stereo imagery based on structural similarity. *Geocarto Int.* **2022**, *37*, 11390–11419. [CrossRef]
13. Alzaghoul, E.; Al-Zoubi, M.B.; Obiedat, R.; Alzaghoul, F. Applying machine learning to DEM raster images. *Technologies* **2021**, *9*, 87. [CrossRef]
14. Nilsson, H.; Pilesjö, P.; Hasan, A.; Persson, A. Dynamic spatio-temporal flow modeling with raster DEMs. *Trans. GIS* **2022**, *26*, 1572–1588. [CrossRef]
15. Dehvari, A.; Heck, R.J. Removing non-ground points from automated photo-based DEM and evaluation of its accuracy with LiDAR DEM. *Comput. Geosci.* **2012**, *43*, 108–117. [CrossRef]
16. DeWitt, J.; Warner, T.; Chirico, P.; Bergstresser, S. Creating high-resolution bare-earth digital elevation models (DEMs) from stereo imagery in an area of densely vegetated deciduous forest using combinations of procedures designed for lidar point cloud filtering. *GIScience Remote Sens.* **2017**, *54*, 552–572. [CrossRef]
17. Sammartano, G.; Spanò, A. DEM Generation based on UAV Photogrammetry Data in Critical Areas. In Proceedings of the GISTAM, Rome, Italy, 26–27 April 2016; pp. 92–98.
18. Axelsson, P. Processing of laser scanner data—Algorithms and applications. *ISPRS J. Photogramm. Remote Sens.* **1999**, *54*, 138–147. [CrossRef]
19. Zhang, K.; Whitman, D. Comparison of Three Algorithms for Filtering Airborne Lidar Data. *Photogramm. Eng. Remote Sens.* **2005**, *71*, 313–324. [CrossRef]
20. Meng, X. A slope-and elevation-based filter to remove non-ground measurements from airborne LIDAR data. *UCGIS Summer Assembly* **2005**, *28*.
21. Ozcan, A.H.; Unsalan, C.; Reinartz, P. Ground filtering and DTM generation from DSM data using probabilistic voting and segmentation. *Int. J. Remote Sens.* **2018**, *39*, 2860–2883. [CrossRef]
22. Sithole, G. Filtering of Laser Altimetry Data using a Slope Adaptive Filter. *Int. Arch. Photogramm. Remote Sens.* **2001**, *XXXIV-3/W4*, 203–210.
23. Vosselman, G. Slope based filtering of laser altimetry data. *IAPRS* **2000**, *XXXIII*, 935–942.
24. Kilian, J.; Haala, N.; Englich, M.; Iii, C. Capture Andevaluation of Airborne Laser Scanner Data. *Int. Arch. Photogramm. Remote Sens.* **2001**, *XXXI*, 383–388.
25. Keqi, Z.; Shu-Ching, C.; Whitman, D.; Mei-Ling, S.; Jianhua, Y.; Chengcui, Z. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 872–882. [CrossRef]
26. Axelsson, P. DEM generation from laser scanner data using adaptive TIN models. *Int. Arch. Photogramm. Remote Sens.* **2000**, *33*, 110–117.

27. Zhang, W.; Qi, J.; Wan, P.; Wang, H.; Xie, D.; Wang, X.; Yan, G. An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation. *Remote Sens.* **2016**, *8*, 501. [CrossRef]
28. Oshio, H.; Yashima, K.; Matsuoka, M. Generating DTM from DSM Using a Conditional GAN in Built-Up Areas. *IEEE Geosci. Remote Sens. Lett.* **2023**, *21*, 2500505. [CrossRef]
29. Sithole, G.; Vosselman, G. Experimental Comparison of Filter Algorithms for Bare-Earth Extraction from Airborne Laser Scanning Point Clouds. *ISPRS J. Photogramm. Remote Sens.* **2004**, *59*, 85–101. [CrossRef]
30. Kasser, M.; Egels, Y. *Digital Photogrammetry*; Taylor & Francis: London, UK, 2002; Volume 11, p. 221.
31. Haralick, R.M.; Sternberg, S.R.; Zhuang, X. Image Analysis Using Mathematical Morphology. *IEEE Trans. Pattern Anal. Mach. Intell.* **1987**, *PAMI-9*, 532–550. [CrossRef]
32. Meng, X.; Wang, L.; Silván-Cárdenas, J.L.; Currit, N. A multi-directional ground filtering algorithm for airborne LIDAR. *ISPRS J. Photogramm. Remote Sens.* **2009**, *64*, 117–124. [CrossRef]
33. Alexander, W.E.; Reeves, D.S.; Gloster, C. Parallel image processing with the block data parallel architecture. *Proc. IEEE* **2002**, *84*, 947–968. [CrossRef]
34. Boykov, Y.Y.; Jolly, M.P. Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In Proceedings of the Eighth IEEE International Conference on Computer Vision, ICCV 2001, Vancouver, BC, Canada, 7–14 July 2001; Volume 101, pp. 105–112.
35. Sternberg, S.R. Grayscale morphology. *Comput. Vis. Graph. Image Process.* **1986**, *35*, 333–355. [CrossRef]
36. Chaki, N.; Shaikh, S.H.; Saeed, K. A comprehensive survey on image binarization techniques. *Explor. Image Bin. Tech.* **2014**, *560*, 5–15.
37. He, L.; Ren, X.; Gao, Q.; Zhao, X.; Yao, B.; Chao, Y. The connected-component labeling problem: A review of state-of-the-art algorithms. *Pattern Recognit.* **2017**, *70*, 25–43. [CrossRef]
38. He, L.; Chao, Y.; Suzuki, K.; Wu, K. Fast connected-component labeling. *Pattern Recognit.* **2009**, *42*, 1977–1987. [CrossRef]
39. Ismail, Z.; Jaafar, J. DEM derived from photogrammetric generated DSM using morphological filter. In Proceedings of the 2013 IEEE 4th Control and System Graduate Research Colloquium, Shah Alam, Malaysia, 20 August 2013; pp. 103–106.
40. Anders, N.; Valente, J.; Masselink, R.; Keesstra, S. Comparing filtering techniques for removing vegetation from UAV-based photogrammetric point clouds. *Drones* **2019**, *3*, 61. [CrossRef]
41. Meng, X.; Currit, N.; Zhao, K. Ground Filtering Algorithms for Airborne LiDAR Data: A Review of Critical Issues. *Remote Sens.* **2010**, *2*, 833–860. [CrossRef]
42. Sulung, P.P.; Soeksmantono, B. Comparative Analysis of DTM Extraction from Airborne LiDAR Point Cloud Data with Adaptive TIN Filter, Cloth Simulation Filter, and Progressive Morphological Filter Methods. *J. Geogr. Gea* **2025**, *25*, 1–14.
43. Koo, S.; Lim, E.T.; Jung, Y.H.; Suk, J.W.; Kim, S.S. Study on applicability of cloth simulation filtering algorithm for segmentation of ground points from drone LiDAR point clouds in mountainous areas. *Korean J. Remote Sens.* **2023**, *39*, 827–835.
44. Li, H.; Ye, C.; Guo, Z.; Wei, R.; Wang, L.; Li, J. A fast progressive TIN densification filtering algorithm for airborne LiDAR data using adjacent surface information. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 12492–12503. [CrossRef]
45. Chen, C.; Guo, J.; Wu, H.; Li, Y.; Shi, B. Performance comparison of filtering algorithms for high-density airborne LiDAR point clouds over complex LandScapes. *Remote Sens.* **2021**, *13*, 2663. [CrossRef]
46. Chai, C.; Zhang, M.; Zuo, T.; Yang, Z.; Liu, Y.; Hou, P.; Wang, Z. *Comparative Study on Point Cloud Filtering Algorithm in Constructing Mining Subsidence Model*; SPIE: Bellingham, WA, USA, 2024; Volume 13223.
47. Patel, H.B.; Singla, J.G. Evaluation and Analysis of Ground Filtering Algorithms for Building Height Estimation on Satellite-Based High-Resolution DEM Data. *J. Indian Soc. Remote Sens.* **2023**, *51*, 661–672. [CrossRef]
48. Hudson, W.D. Correct formulation of the kappa coefficient of agreement. *Photogramm. Eng. Remote Sens.* **1987**, *53*, 421–422.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.