

## Article

# Drone Multiline Light Detection and Ranging Data Filtering in Coastal Salt Marshes Using Extreme Gradient Boosting Model

Xixiu Wu <sup>1</sup>, Kai Tan <sup>2</sup>, Shuai Liu <sup>2,\*</sup> , Feng Wang <sup>1</sup>, Pengjie Tao <sup>3</sup> , Yanjun Wang <sup>4</sup>  and Xiaolong Cheng <sup>5</sup> 

<sup>1</sup> Guangzhou Urban Planning and Design Survey Research Institute, Guangzhou 510060, China; wuxixiu@gzpi.com.cn (X.W.); wangfeng@gzpi.com.cn (F.W.)

<sup>2</sup> State Key Laboratory of Estuarine and Coastal Research, East China Normal University, Shanghai 200241, China; ktan@sklec.ecnu.edu.cn

<sup>3</sup> School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China; pjtao@whu.edu.cn

<sup>4</sup> Hunan Provincial Key Laboratory of Geo-Information Engineering in Surveying, Mapping and Remote Sensing, Hunan University of Science and Technology, Xiangtan 411201, China; wangyanjun@hnust.edu.cn

<sup>5</sup> College of Civil and Surveying and Mapping Engineering, Jiangxi University of Science and Technology, Ganzhou 341000, China; chengxiaolong@jxust.edu.cn

\* Correspondence: liushuai\_820@stu.ecnu.edu.cn

**Abstract:** Quantitatively characterizing coastal salt-marsh terrains and the corresponding spatiotemporal changes are crucial for formulating comprehensive management plans and clarifying the dynamic carbon evolution. Multiline light detection and ranging (LiDAR) exhibits great capability for terrain measuring for salt marshes with strong penetration performance and a new scanning mode. The prerequisite to obtaining the high-precision terrain requires accurate filtering of the salt-marsh vegetation points from the ground/mudflat ones in the multiline LiDAR data. In this study, a new alternative salt-marsh vegetation point-cloud filtering method is proposed for drone multiline LiDAR based on the extreme gradient boosting (i.e., XGBoost) model. According to the basic principle that vegetation and the ground exhibit different geometric and radiometric characteristics, the XGBoost is constructed to model the relationships of point categories with a series of selected basic geometric and radiometric metrics (i.e., distance, scan angle, elevation, normal vectors, and intensity), where absent instantaneous scan geometry (i.e., distance and scan angle) for each point is accurately estimated according to the scanning principles and point-cloud spatial distribution characteristics of drone multiline LiDAR. Based on the constructed model, the combination of the selected features can accurately and intelligently predict the category of each point. The proposed method is tested in a coastal salt marsh in Shanghai, China by a drone 16-line LiDAR system. The results demonstrate that the averaged AUC and G-mean values of the proposed method are 0.9111 and 0.9063, respectively. The proposed method exhibits enhanced applicability and versatility and outperforms the traditional and other machine-learning methods in different areas with varying topography and vegetation-growth status, which shows promising potential for point-cloud filtering and classification, particularly in extreme environments where the terrains, land covers, and point-cloud distributions are highly complicated.

**Keywords:** drone; multiline LiDAR; salt marshes; point-cloud filtering; machine learning



**Citation:** Wu, X.; Tan, K.; Liu, S.; Wang, F.; Tao, P.; Wang, Y.; Cheng, X. Drone Multiline Light Detection and Ranging Data Filtering in Coastal Salt Marshes Using Extreme Gradient Boosting Model. *Drones* **2024**, *8*, 13. <https://doi.org/10.3390/drones8010013>

Academic Editor: Pablo Rodríguez-González

Received: 15 November 2023

Revised: 19 December 2023

Accepted: 22 December 2023

Published: 4 January 2024



**Copyright:** © 2024 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

Salt marshes are geolocated at the transitional coastal zones between the land and sea. Due to the distinctive geographic environment and location, salt marshes are periodically submerged by the tides [1] and have a rich variety of flora and fauna resources [2]. As such, salt marshes are generally recognized as a major contributor to global coastal blue carbon ecosystems and possess enormous ecological and economic values [3,4]. Geologically, salt marshes can naturally respond to sea-level rise by the physical accumulation of mineral and

biogenic sediments [5,6]. Consequently, terrains and landforms can contribute to the quantitative investigation of the impact mechanism for salt-marsh dynamics, geomorphology, and sedimentation processes, as well as the acquisition of salt-marsh biomass [7–9]. Accurately obtaining the terrain data and the corresponding spatiotemporal evolutions can provide parameterization or complements for the investigation of the carbon–water–sediment dynamic exchange process and feedback mechanisms for salt marshes [10]. However, dense vegetation coverage, periodic water accumulation, and muddy circumstances often make in situ traditional terrain-measuring techniques infeasible. Technological advancements in optical remote sensing have introduced a number of remarkable alternative techniques for the investigation of salt-marsh terrains in a noncontact way [11–13]. Passive remote-sensing technologies do not directly record the terrain. As such, some strategies or methods, e.g., oblique photography [14], stereo image pairs [15], and waterline detection [12], are needed for terrain derivation with the assistance of ground truth observation data. However, optical remote sensing is a passive technique and the acquired images are vulnerable to the environment (e.g., sunlight, the atmosphere, and clouds). Additionally, optical remote sensing can only provide two-dimensional information on the vegetation canopies. Only the terrains of the regions without vegetation or with sparse vegetation can be satisfactorily obtained [16–19].

LiDAR (light detection and ranging) can acquire the geometric and radiometric data of each scanned point simultaneously by actively emitting and receiving lasers. Compared with optical remote sensing, LiDAR is characterized by strong penetrability, high-resolution data quality, and three-dimensional data acquisition [20–22]. However, traditional LiDAR systems commonly employ a single emitter and receiver to detect targets, i.e., laser beams are emitted successively at a very short constant time interval. Conversely, multiline LiDAR (e.g., 4-, 8-, 16-, 32-, 64-, and 128-line), a novel technology that has boomed in recent years, can emit multiple laser shots simultaneously to improve data accuracy and reliability [23]. Multiline LiDAR is an “upgraded version” of traditional LiDAR and has qualitative improvements in data dimension and scene restoration. Multiline LiDAR can realize high-precision 3D environment modeling and detect obstacles at different heights, which is generally used in the field of unmanned driving or drone detection. Drone multiline LiDAR enables more effective data collection and penetration in dense vegetation regions and shows great potential in terrain measuring for salt marshes [24].

The prerequisite is to filter the drone multiline LiDAR point clouds for terrain derivation in salt marshes, which means accurately classifying the raw data into two categories, that is, ground (i.e., mudflat) points and nonground (i.e., vegetation) ones [25]. Massive advanced filtering algorithms have been developed, including progressive morphological filtering (PMF) [26], slope-based filtering (SF) [27], and cloth-simulation filtering (CSF) [28]. The basic filtering strategies of these methods mostly lie in the geometric differences between ground and nonground points. However, these filtering algorithms cannot be robustly and accurately applied in salt marshes due to the tortuous creeks being heterogeneous in depth and the dense vegetation varying in structures [24]. In addition to the geometric differences, vegetation is composed of materials and water contents different from that of the mudflat and has totally different reflectance properties [24]. Therefore, vegetation can be filtered by the LiDAR-intensity data which contain the spectral properties of the scanned points. However, multiple factors complicate the original intensity, and these effects must be eliminated. Considerable progress has been achieved in LiDAR-intensity correction in the past decade, and many advanced models have been developed [29–32]. Nevertheless, different LiDAR instruments must be individually calibrated, and the corrected intensity data at a single-wavelength laser cannot separate all vegetation points. Moreover, strong specular reflections typically occur over wet or water-dominated salt marshes, increasing the difficulties in intensity correction [33]. Hence, intensity data are a potential metric for vegetation separating but intensity correction is a big challenge. A number of previous studies have demonstrated that a feasible optimization solution for filtering is to combine the intensity and geometric data [34].

In addition to the traditional filtering methods based on either geometric or intensity data, another new potential strategy is machine-learning-based methods, e.g., adaptive boosting (AdaBoost), extreme gradient boosting (XGBoost), and random forest (RF) [35]. Machine-learning-based methods can automate feature learning, handle high-dimensional data, and deal with nonlinear relationships, which are suitable for various scenarios and different filtering tasks [36]. In particular, the XGBoost can efficiently process massive data samples through noncore computation, which demonstrates superior classification performance and effectiveness compared to other commonly used machine-learning methods and shows great potential in 3D point-cloud processing [37]. However, the performance of the XGBoost for filtering salt-marsh vegetation remains unknown, since the scanning mode of drone multiline LiDAR is totally different from the traditional LiDAR; the salt-marsh environment is extremely complicated.

In this study, we propose a new alternative method to filter the drone multiline LiDAR point cloud of coastal salt-marsh vegetation based on the XGBoost model. The fundamental principle of the proposed method is to characterize the specific relationship of the point attributes (mudflat or vegetation) and their corresponding geometric and radiometric characteristics by constructing an XGBoost model. In the model, a total of five fundamental geometric and radiometric quantities of each point, i.e., scan angle, distance, elevation, normal vectors, and original intensity, are used as independent variables, and point attributes are used as dependent variables. The instantaneous scanning geometry (i.e., scan angle and distance) of each point is estimated based on the scanning principles of the drone multiline LiDAR system. Different from the geometric data-based method that needs to manually set/derive a number of parameters and perform extensive 3D operations, the intensity-data-based methods that need to estimate the corresponding model parameters after establishing a specific mathematical correction model, or the combination methods that need to use the geometric and the radiometric information successively by several procedures, the intensity correction and the combination of the geometric and the radiometric data are totally realized by the XGBoost model. The major innovations include (1) a novel method proposed to accurately estimate the instantaneous scan geometry of the drone multiline LiDAR system and (2) a new point-cloud filtering method for salt-marsh vegetation based on XGBoost is proposed, which can avoid the complex intensity-correction process and internally combines the geometric and radiometric data for point-cloud classification.

## 2. Methodology

### 2.1. Architecture Overview

A total of  $N$  original points collected by drone multiline LiDAR are represented by  $P = \{P_i(X_i, Y_i, Z_i, I_i, T_i), i = 1, 2, \dots, N\}$ , where  $(X_i, Y_i, Z_i)$  are the 3D coordinates,  $I_i$  is the original intensity, and  $T_i$  is the GNSS time. The flight height ( $H$ ) is provided by the drone system and the take-off point elevation ( $z_0$ ) is measured by real-time kinematic (RTK). Four major procedures are included in the proposed method, as shown by the architecture overview (Figure 1). First, the absent instantaneous distance  $d_i$  and scan angle  $\theta_i$  of each point is accurately estimated based on the 3D coordinates, GNSS time, flight height, and take-off point elevation. Meanwhile, the surface normal vectors  $(\mu_i, \nu_i, \omega_i)$  are derived by the best-fitting plane around the interest point using the 3D coordinates of the nearest neighbor points by the least squares adjustment method [38]. Second, a feature set including the distance ( $d_i$ ), scan angle ( $\theta_i$ ), elevation ( $z_i$ ), normal vectors  $(\mu_i, \nu_i, \omega_i)$ , and original intensity ( $I_i$ ) is constructed for each point. Third, the XGBoost is trained and optimized using the training and test sets, whereas the model performance is assessed by the validation datasets. Finally, the constructed model is popularized to filter the entire point cloud of the study site.

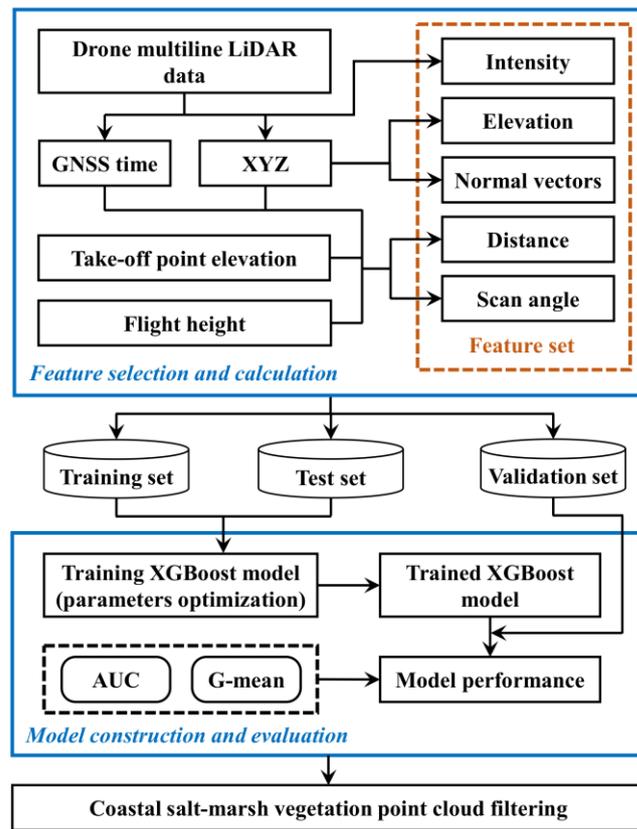


Figure 1. Architecture overview of the proposed method.

### 2.2. XGBoost Algorithm

XGBoost improves classification accuracy by building an ensemble of multiple weak classifiers [39]. The fundamental principle of XGBoost is to fit the residuals by continuously iterating and to update the model parameters by optimizing the objective function (Figure 2).

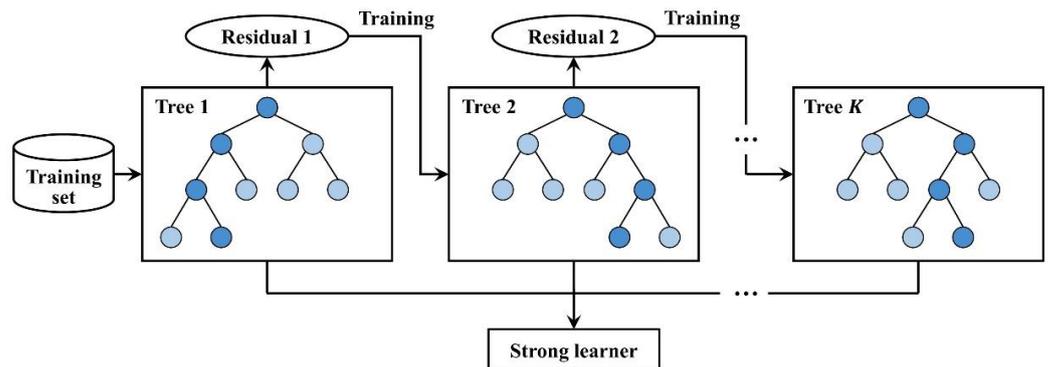


Figure 2. Structure of XGBoost algorithm.

The goal of XGBoost at each iteration is to eliminate the residual of the previous iteration. The process starts from the first tree until the optimization of the  $K$ -th tree is completed. The details can be described as follows:

$$\begin{cases} \hat{y}_i^{(0)} = 0 \\ \hat{y}_i^{(1)} = f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i) \\ \hat{y}_i^{(2)} = f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i) \\ \dots \\ \hat{y}_i^{(K)} = \sum_{k=1}^K f_k(x_i) = \hat{y}_i^{(K-1)} + f_K(x_i) \end{cases} \quad (1)$$

where  $x_i$  is the training sample,  $\hat{y}_i^{(K)}$  is the final model prediction value,  $\hat{y}_i^{(K-1)}$  is the prediction of the model after  $K - 1$  iterations, and  $f_k(x_i)$  is the prediction of the  $k$ -th tree.

XGBoost uses a gradient-boosting strategy to optimize the objective function. By using a second-order Taylor expansion of the loss function and introducing a regularization term, XGBoost effectively improves model accuracy while avoiding the risk of overfitting. However, it requires considerable computational resources for large datasets.

### 2.3. Feature Selection and Derivation

#### 2.3.1. Feature Selection

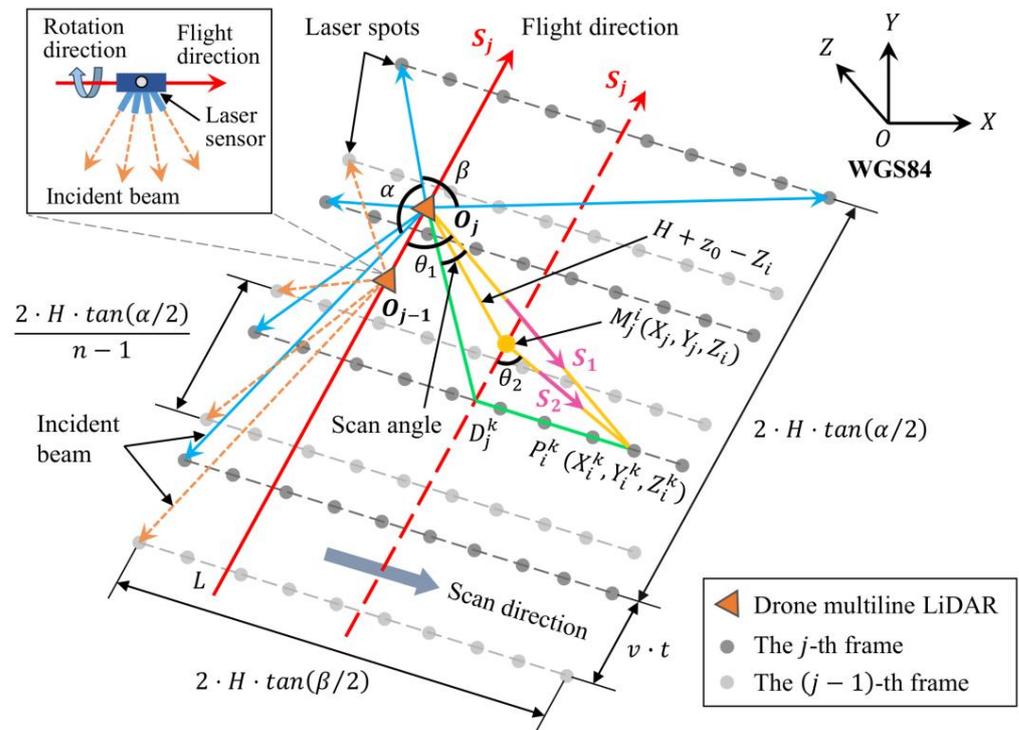
Selecting appropriate features for XGBoost can improve model performance and accuracy while reducing computational complexity and overfitting risk [40]. In the proposed method, the intensity, incidence angle, distance, elevation, and normal vectors are used as input features to build an XGBoost filtering model. Intensity demonstrates the reflectance of the target surface and can discriminate different types of objects. The incidence angle and distance are the predominant factors that affect intensity. By imputing intensity, distance, and scan angle as features in the XGBoost model, the influences of scan geometry on intensity can be fully eliminated and the traditionally complex intensity-correction processes are avoided. However, only the single-wavelength intensity data are not able to separate all vegetation points. The orientations of the vegetation points are irregular, whereas those of the ground points are nearly perpendicular. Additionally, the vegetation is generally higher than the mudflat. As such, the normal vectors and elevation (i.e., the  $z$  coordinate), which indicate the surface orientation and height of each point, respectively, are used to compensate for the insufficiency of the intensity data.

#### 2.3.2. Derivation of Scan Angle and Distance

In practice, the user-provided LAS file usually does not involve the 3D coordinates of the instrument and each point at the instantaneous scanning time [41]. Consequently, neither the incidence angle nor the distance can be acquired directly. Due to constant flight height and flat salt-marsh terrain, the incidence angle can be approximately substituted by the scan angle [24,41]. A new strategy is realized in [24] to restore the scan angle and distance of each point for drone LiDAR based on the GNSS time and flight height. In the method, the projection point  $M_j^i$  of the instrument in a short time interval is approximately considered the projection points of all the  $n$  emitters of the multiline LiDAR, i.e., the projection points of the  $n$  emitters coincide. However, each emitter should have an individual projection point. For example, the projection point of the  $k$ -th ( $k = 1, 2, \dots, n$ ) emitter in the  $j$  frame is  $D_j^k$ , whereas  $M_j^i$  is the projection point of the instrument (or the midpoint of the projection points of the  $n$  emitters). As such, the scan angle of a point  $P_i^k \left( X_i^k, Y_i^k, Z_i^k \right)$  scanned by the  $k$ -th emitter in the  $j$  frame is  $\angle D_j^k O_j P_i^k$  rather than  $\angle M_j^i O_j P_i^k$ . The details for deriving  $\angle D_j^k O_j P_i^k$  are introduced as follows.

Multiline LiDAR takes frames as scanning units (Figure 3). The rotation axis of the laser sensor is aligned with the flight direction of the drone. In the plane where the rotation axis is located, the laser sensor simultaneously emits a group of beams in different directions with multiple included angles with the rotation axis. The included angles are preset by the manufacturer and keep constant all the time. Then, the laser sensor undergoes  $360^\circ$  rotational scanning around the rotation axis. For a  $n$ -line LiDAR with a frequency of  $f$ , the  $n$  emitters simultaneously emit a group of  $n$  lasers in a moment every  $t$  seconds, i.e.,

the time interval between two neighbor frames is  $t$  where  $t = 1/f$ . In each frame, there are a total of  $n$  scan lines, and these scan lines are parallel to each other. There are  $f$  frames and  $n \cdot f$  scan lines per second. If the flight speed of the drone is  $v$  m/s, then the distance between two neighbor frames is  $v \cdot t$ . If  $N_t$  points are scanned per second, then each frame contains  $N_t \cdot t$  points and each scan line contains  $N_t \cdot t/n$  points. In a single frame, the forward and side fields of view of the LiDAR are  $\alpha$  and  $\beta$ , respectively. The length of a frame is  $2 \cdot H \cdot \tan(\alpha/2)$ , and the width of a frame (i.e., swath) is  $2 \cdot H \cdot \tan(\beta/2)$ , where  $H$  is the flight height. The distance between two neighbor scanning lines in a certain frame is  $2 \cdot H \cdot \tan(\alpha/2)/(n - 1)$ , and the distance between two neighbor points (i.e., spatial resolution) on a certain scan line is  $2 \cdot H \cdot \tan(\beta/2)/[N_t \cdot t/n - 1]$ . Let  $j$  be the frame number and  $k$  ( $k = 1, 2, \dots, n$ ) be the scan line/emitter number.  $O_j$  is the position of the drone in the  $j$  frame. We can segment the point clouds into a group of individual frames by the GNSS time of each point. However, it is a challenge to further extract each scan line in an individual frame, and, thus, the position of  $D_j^k$  cannot be directly obtained by the points in a single scan line. This condition means that  $\angle D_j^k O_j P_i^k$  cannot be directly calculated in  $\Delta D_j^k O_j P_i^k$ .



**Figure 3.** Geometric relations during drone  $n$ -line LiDAR data collection ( $n$  is 4 for the case).

The obtained point clouds  $P_i = (X_i, Y_i, Z_i, I_i, T_i)$ ,  $i = 1, 2, \dots, N$ , are divided into  $N_f$  individual frames at a uniform interval of  $t$  ( $t = 1/f$ ) on the basis of the GNSS time ( $T_i$ ). In the  $j$ -th frame, a total of  $N_j$  ( $N_j = N_t \cdot t$ ) points are acquired. Let  $M_j^i$  be the center of the  $N_j$  points. The plane coordinates  $(X_j, Y_j)$  of  $M_j^i$  can be calculated as follows [24]:

$$\begin{cases} X_j = \frac{\sum_{l=0}^{N_j} (X_l^i)}{N_j} \\ Y_j = \frac{\sum_{l=0}^{N_j} (Y_l^i)}{N_j} \end{cases} \quad (2)$$

where  $X_l^i$  and  $Y_l^i$  are the  $x$  and  $y$  coordinates of the points.  $M_j^i$  is the projection point of the drone platform at the  $j$ -th frame. Therefore, the coordinates of the  $n$  laser scanners are  $(X_j, Y_j, H + z_0)$ .

Let the flight direction vector of the drone at the  $j$ -th frame be  $S_j = (u_j, v_j, w_j)$ .  $S_j$  can be calculated by fitting the coordinates of the drone in the neighbor frames of the  $j$ -th frame by the least squares. Let  $\angle LO_j P_i^k$  and  $\angle D_j^k M_j^i P_i^k$  be  $\theta_1$  and  $\theta_2$ , respectively. The vector of the scanning line  $O_j P_i^k$  is  $S_1 = (X_i^k - X_j, Y_i^k - Y_j, Z_i^k - H - z_0)$ . Then, the included angle  $\theta_1$  between  $O_j P_i^k$  and the drone flight direction line can be obtained as follows.

$$\theta_1 = \cos^{-1} \left( \frac{S_j \cdot S_1}{|S_j| \cdot |S_1|} \right) = \cos^{-1} \left[ \frac{u_j (X_i^k - X_j) + v_j (Y_i^k - Y_j) + w_j (Z_i^k - H - z_0)}{\sqrt{u_j^2 + v_j^2 + w_j^2} \cdot \sqrt{(X_i^k - X_j)^2 + (Y_i^k - Y_j)^2 + (Z_i^k - H - z_0)^2}} \right] \quad (3)$$

Similar to  $\theta_1$ ,  $\theta_2$  can be derived as follows.

$$\theta_2 = \cos^{-1} \left( \frac{S_j \cdot S_2}{|S_j| \cdot |S_2|} \right) = \cos^{-1} \left[ \frac{u_j (X_i^k - X_j) + v_j (Y_i^k - Y_j)}{\sqrt{u_j^2 + v_j^2 + w_j^2} \cdot \sqrt{(X_i^k - X_j)^2 + (Y_i^k - Y_j)^2}} \right] \quad (4)$$

where  $S_2 = (X_i^k - X_j, Y_i^k - Y_j, 0)$  is the vector of the line  $M_j^i P_i^k$ .

According to the scanning principles of drone multiline LiDAR, the  $n$  emitters are rotated and scanned at a fixed angle step around the drone flight direction line in each frame (Figure 3). Therefore, the included angles between each individual incident beam and the drone flight direction line are equal for each point in a certain scan line, i.e.,  $\angle LO_j D_j^k = \angle LO_j P_i^k = \theta_1$ . Thus,  $\angle D_j^k O_j M_j^i = \pi/2 - \angle LO_j D_j^k = \pi/2 - \theta_1$ . In  $\Delta D_j^k O_j M_j^i$  (Figure 3), the following relationships can be obtained.

$$\begin{cases} O_j D_j^k = (H + z_0 - Z_i^k) / \cos \angle D_j^k O_j M_j^i = (H + z_0 - Z_i^k) / \sin \theta_1 \\ D_j^k M_j^i = (H + z_0 - Z_i^k) \cdot \tan \angle D_j^k O_j M_j^i = (H + z_0 - Z_i^k) \cdot \cot \theta_1 \end{cases} \quad (5)$$

In  $\Delta D_j^k M_j^i P_i^k$  (Figure 3),  $D_j^k P_i^k$  can be obtained according to the law of cosines.

$$D_j^k P_i^k{}^2 = D_j^k M_j^i{}^2 + M_j^i P_i^k{}^2 - 2 \cdot D_j^k M_j^i \cdot M_j^i P_i^k \cdot \cos \theta_2 \quad (6)$$

where  $M_j^i P_i^k = \sqrt{(X_i^k - X_j)^2 + (Y_i^k - Y_j)^2}$ . It is worth noting that the scan lines are not strictly perpendicular to the flight direction of the drone since the drone is in uniform motion. However, the laser sensor has an extremely high scan rate, allowing the effect of this factor to be neglected, and the scan lines can be approximated to perpendicular to the flight direction of the drone, i.e.,  $\angle P_i^k D_j^k M_j^i = 90^\circ$ . In Equation (6),  $\theta_2$  and  $D_j^k M_j^i$  can be obtained by Equations (4) and (5), respectively. In  $\Delta D_j^k O_j P_i^k$  (Figure 3), the scan angle and distance for an arbitrary point  $P_i^k$  can be obtained by substituting Equations (5) and (6) into Equation (7),

$$\begin{cases} \theta(P_i^k) = \angle D_j^k O_j P_i^k = \cos^{-1} \left( \frac{O_j D_j^k{}^2 + O_j P_i^k{}^2 - D_j^k P_i^k{}^2}{2 \cdot O_j D_j^k \cdot O_j P_i^k} \right) \\ d(P_i^k) = O_j P_i^k = \sqrt{(X_i^k - X_j)^2 + (Y_i^k - Y_j)^2 + (Z_i^k - H - z_0)^2} \end{cases} \quad (7)$$

Incident angle (scan angle) and distance are the predominant influencing factors of UAV LiDAR-intensity data. Many advanced intensity-correction models have been developed to derive a value that is equal to or associated with the target reflectance, including theoretical, empirical, and reference-target models [29–32]. However, these methods commonly require independent correction for the incident angle (scan angle) and the distance individually by deriving the specific complex mathematical relations between intensity and distance/incident angle (scan angle) using standard Lambertian or naturally

homogeneous targets. By simultaneously inputting scan angle, distance, and intensity into the XGBoost model, it can automatically capture the nonlinear relationship between the intensity and target properties, considering the influences of distance and incident angle (scan angle). The proposed method can avoid the traditional complex intensity-correction process and can be efficiently used for target classification.

#### 2.4. Accuracy Evaluation

The filtering accuracy is usually assessed using the type I/II error and the total error [42]. However, only a few ground points can be acquired in the salt marshes. Thus, the numbers of ground and vegetation points are extremely unbalanced. Under this circumstance, several unreasonable results could appear if the error-evaluation standard proposed by ISPRS is adopted. The ROC (receiver operating characteristic) curve [43] provides a novel way to measure the accuracy performance. This curve uses the false-positive rate ( $FPR = FP/(TN + FP)$ ) as the  $x$ -axis and the true-positive rate ( $TPR = TP/(TP + FN)$ ) as the  $y$ -axis. By adjusting the classification threshold of the model, the ROC curve can be obtained by multiple sets of  $FPR$  and  $TPR$ . In this study, the AUC [44] (i.e., area under the ROC curve) and G-mean [45] are applied to quantitatively assess the filtering accuracy. These two metrics are appropriate for accuracy evaluation for imbalanced samples. The G-mean is defined as

$$\text{G-mean} = \sqrt{\frac{TN}{TN + FP} \times \frac{TP}{TP + FN}} \quad (8)$$

where  $TP$  is the number of ground points correctly classified as ground points,  $TN$  is the number of vegetation points correctly classified as vegetation points,  $FP$  is the number of vegetation points misclassified as ground points, and  $FN$  is the number of ground points misclassified as vegetation points. The values of AUC and G-mean range from 0 to 1, and large values indicate a better filtering result. G-mean provides an overall measure of the classification results for both the ground and vegetation points. The G-mean is close to 1 only when both the ground and vegetation points are satisfactorily classified.

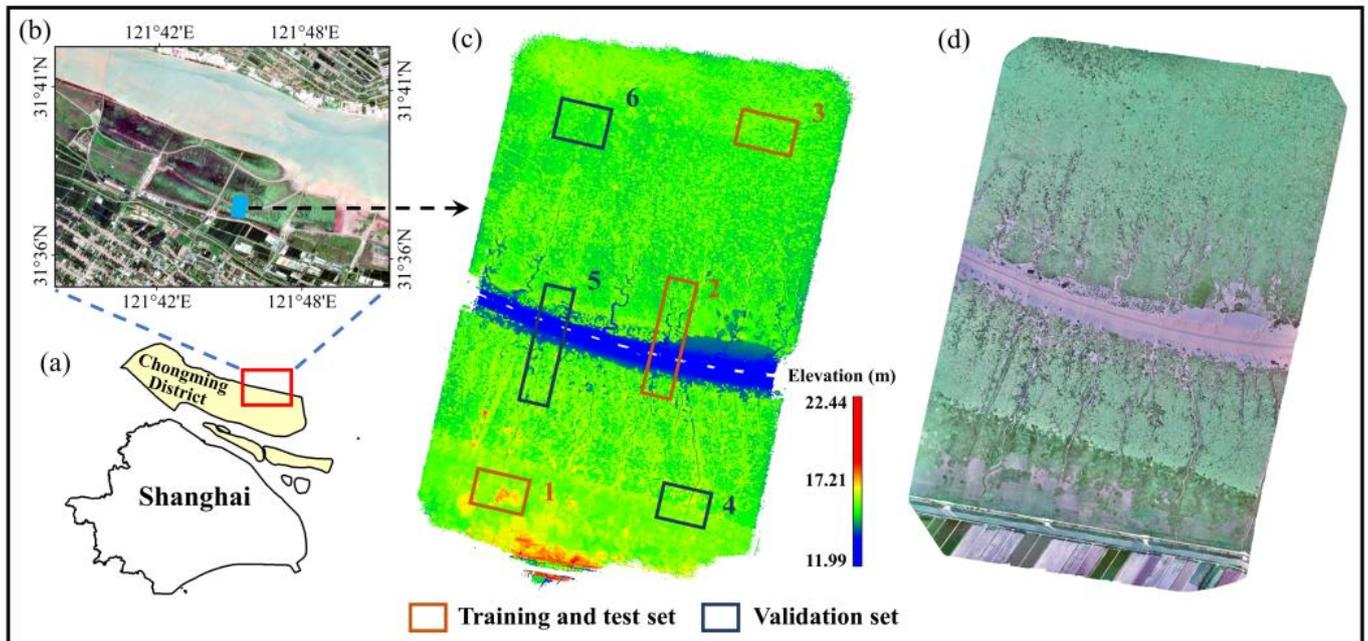
### 3. Experiments and Results

#### 3.1. Data and Materials

In this study, a salt marsh (Figure 4) on Chongming Island, China, was selected to analyze the filtering performance of the proposed method. The vegetation in this salt marsh includes *Phragmites australis* (PA), *Scirpus mariqueter* (SM), and *Spartina alterniflora* (SA). The salt marsh is approximately with an area of 2.6 km<sup>2</sup>. A wide creek is centrally located within the salt marsh, which branches numerous narrow and multilevel creeks. The vegetation is densely distributed among the entire salt marsh, with an average height of 1–2 m. Only a few regions near the central wide creek are bare mudflats.

A “hummingbird” genius multiline LiDAR, equipped with an R-Fans-16 laser scanner capable of simultaneously emitting 16 laser beams, was mounted on the “ZR-M66” drone platform to collect the point clouds of the selected salt marsh on 17 August 2019. The wavelength and maximum scanning distance of the emitted laser are 905 nm (near-infrared) and 250 m, respectively. The field of view of the R-Fans-16 laser scanner is 360° × 30°. The pulse repetition frequency is 320 kHz, and the scanning frequency is 5–20 Hz (5 Hz in this study). The R-Fans-16 laser scanner can receive multiple echoes, ensuring that it can better penetrate the salt-marsh vegetation and obtain high-precision ground data. The flight height during data collection was 80 m, whereas the drone moved at a uniform speed of 7 m/s. The obtained LAS-format data for each point were mainly constituted by the elements of 3D coordinates, GNSS time, and original intensity. The horizontal coordinates ( $x$  and  $y$ ) and elevation ( $z$ ) referred to the WGS84 coordinate system and geodetic height, respectively. The point-clouds preprocessing mainly included outliers and noises removal, which were accomplished by CloudCompare software. A total of 243,098,388 points were obtained for the study site, with overall horizontal and vertical accuracies of 0.10 and 0.15 m, respectively. The acquired point clouds were used to extract the fairy circles in [33]. In this

study, the aim is to extract the vegetation points rather than the fairy circles. Additionally, an orthophoto of the study site was acquired by a Sony RX1R II camera (Figure 4d). More detailed information on the study site and data collection materials can be referred to in [24].



**Figure 4.** (a,b) Location of the study site, (c) drone multiline LiDAR point clouds, (d) Orthophoto.

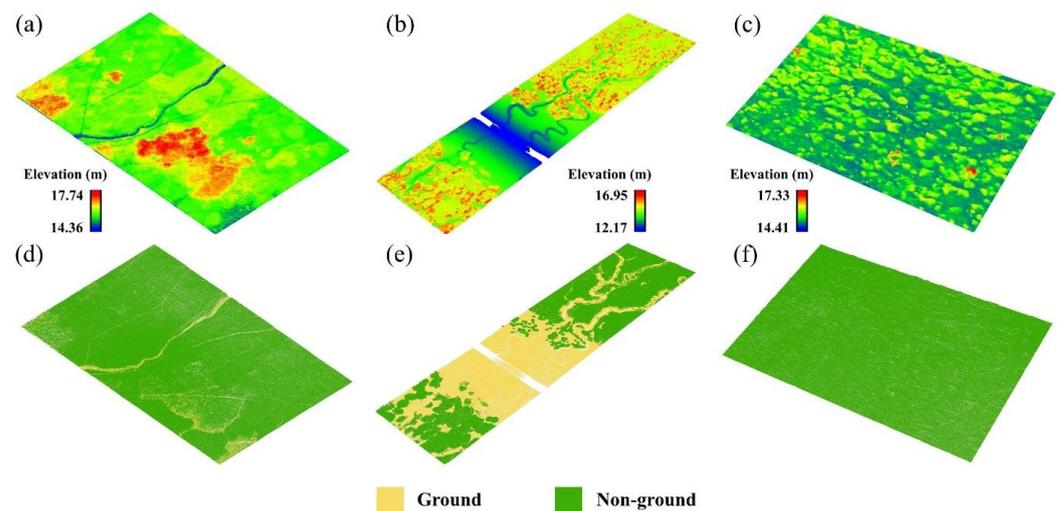
The species and densities of the vegetation and the salt-marsh topography can influence the filtering accuracy and reliability. In this study, three representative subregions (orange boxes in Figure 4c) with varying vegetation and topographic characteristics (Table 1) from the study site were selected to build a generalized XGBoost model. The compositions of the point clouds of the three regions were randomly divided into two subsets according to a 7:3 ratio. Specifically, 70% of the point clouds served as the training set for model training, and 30% of the point clouds acted as the test set for model-performance evaluation. The XGBoost model training includes a number of hyperparameter settings, among which the main parameters include *n\_estimators*, *lambda*, *gamma*, *colsample\_bytree*, *learning\_rate*, *max\_depth*, *subsample*, and *min\_child\_weight*. The hyperparameters were optimized by a grid search method which is widely used for machine-learning hyperparameter optimization. The best parameters of the XGBoost model were obtained when the highest accuracy was achieved on the test set. Merely observing the accuracy performance of the test set cannot demonstrate the generalization ability of the constructed XGBoost model. Therefore, we selected three additional regions with different vegetation and topographic characteristics (Table 1) from the study site (blue boxes in Figure 4c) as the validation set to further test the constructed model. The XGBoost model was trained, tested, and validated using the Scikit-learn library in Python in this study. The reference data for model evaluation were gathered manually, aided by the orthophotos to guarantee precision.

**Table 1.** Information on the training, test, and validation sets.

Dataset	Point Number	Size	Features
Training and test set	1	5,410,276	250 × 160 m (1) Vegetation: PA, SA, and SM, highly dense (2) Several intertidal creeks
	2	5,260,976	130 × 510 m (1) Vegetation: SA and SM, relatively dense (2) Multiple intertidal creeks and a small part of bare mudflat
	3	5,201,640	260 × 170 m (1) Vegetation: SA, highly dense (2) No intertidal creek
Validation set	4	4,780,134	250 × 180 m (1) Vegetation: PA and SA, relatively dense (2) Several intertidal creeks
	5	4,700,965	120 × 490 m (1) Vegetation: SA and SM, relatively dense (2) Multiple intertidal creeks and a small part of bare mudflat
	6	5,147,804	240 × 195 m (1) Vegetation: SA, relatively dense (2) No intertidal creek

### 3.2. Model Training and Testing Results

The separating results of the training and test regions are shown in Figure 5. Visually, the ground points can be effectively separated from various types of vegetation points for Regions 1, 2, and 3; only a small portion was recognized as vegetation points. The intertidal creeks in Regions 1 and 2 are tortuous and the depths vary considerably; nevertheless, they were satisfactorily recognized as ground points. Although the vegetation is highly dense and only a few ground points were obtained for Region 3, the XGBoost model can robustly achieve satisfactory filtering regardless of the imbalance of the ground and vegetation points. The AUC and G-mean were 0.9536 and 0.9528 for the training set, and 0.9404 and 0.9391 for the test set, respectively. The results indicated that the XGBoost can achieve a high accuracy in separating point clouds with varying vegetation types and terrain characteristics.

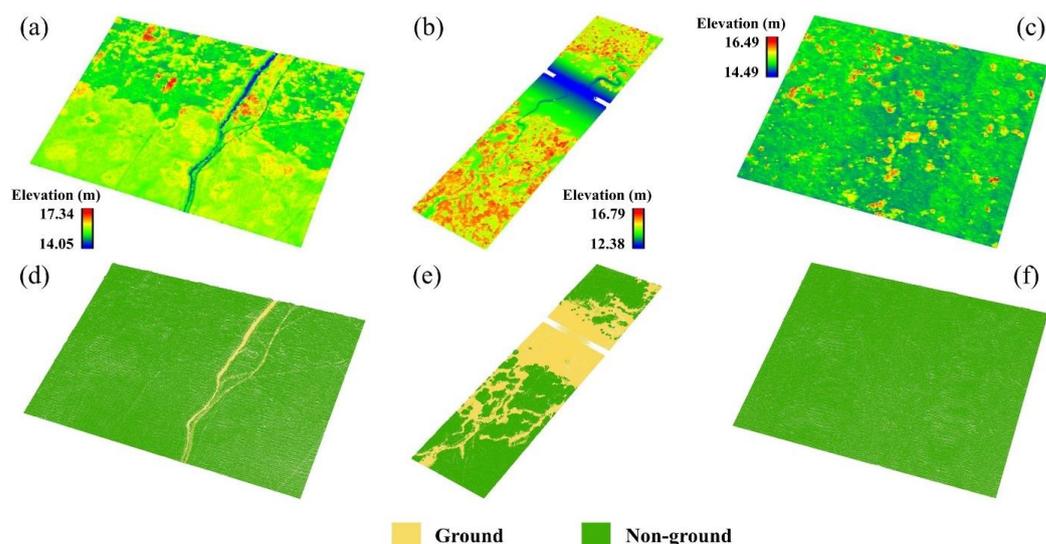


**Figure 5.** Separating results of the training and test sets by the XGBoost model. (a–c) Points are colored by elevation for Regions 1, 2, and 3, respectively. (d–f) Separating results for Regions 1, 2, and 3, respectively.

### 3.3. Model-Validation Results

The trained XGBoost model was migrated to the three validation regions (Figure 6 and Table 2). Obviously, the trained XGBoost model effectively discriminated vegetation points from ground points in all three validation regions, demonstrating that the trained XGBoost model can be robustly extended to different kinds of regions with satisfactory

filtering accuracy. The AUC and G-mean values of the XGBoost model for Regions 4, 5, and 6 were 0.9191 and 0.9158, 0.9535 and 0.9534, and 0.8607 and 0.8496, respectively. The highest accuracy was achieved in Region 5, because this region is covered by bare mudflats and low/sparse vegetation. In contrast, the filtering performance was not so satisfactory for Region 6 compared to the other two validation regions. This phenomenon was primarily attributed to the very dense vegetation coverage in Region 6, resulting in a sparse acquisition of ground points and difficulty in filtering. Additionally, manual separation of the ground and vegetation points in dense vegetation coverage regions is challenging even with the assistance of the orthophoto. Some errors existed in the reference data and this would lead to the unfair accuracy evaluation of Region 6.



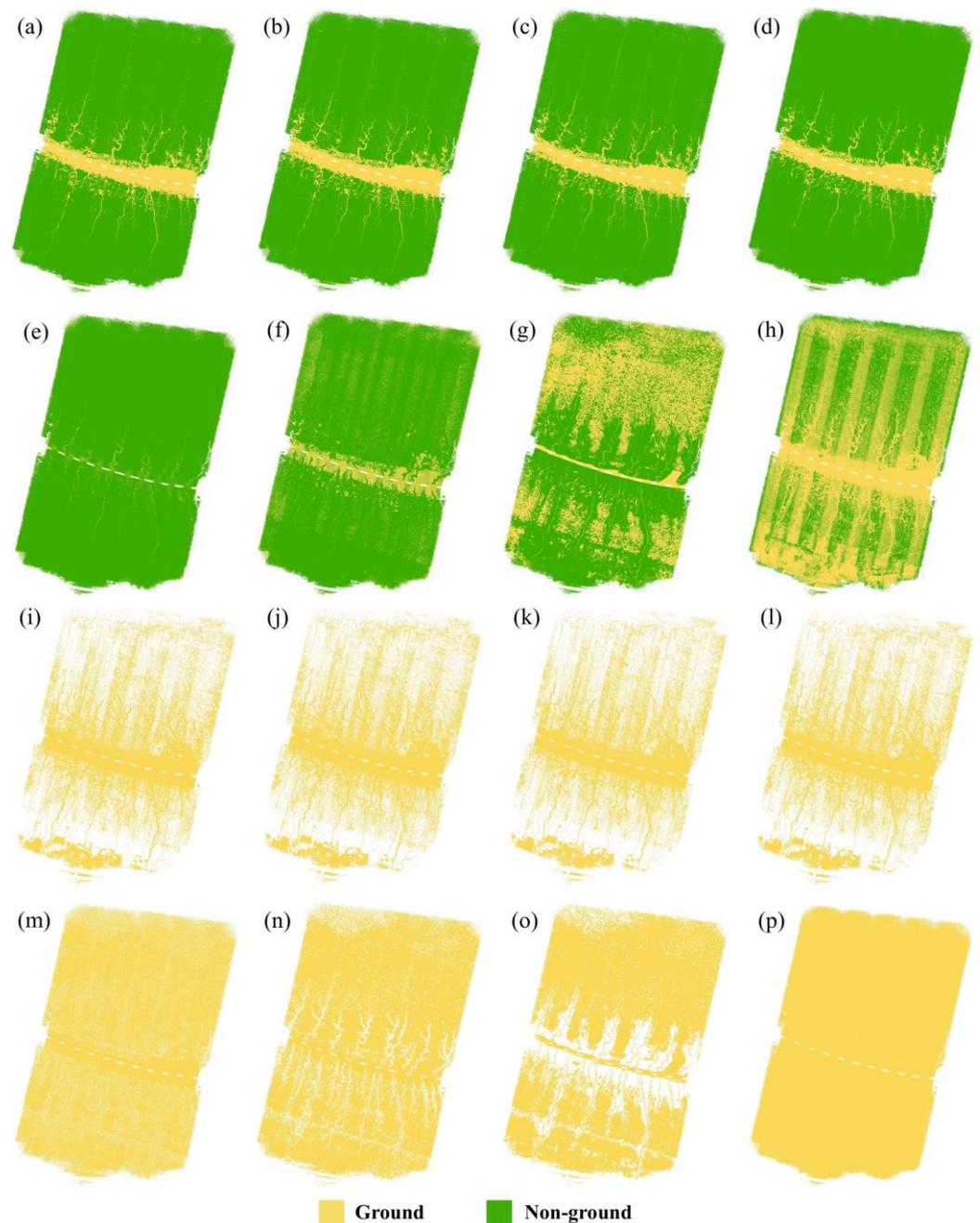
**Figure 6.** Separating results of the validation set by the XGBoost model. (a–c) Points are colored by elevation for Regions 4, 5, and 6, respectively. (d–f) Separating results for Regions 4, 5, and 6, respectively.

**Table 2.** AUC and G-means of the validation set.

	Region 4	Region 5	Region 6
AUC	0.9191	0.9535	0.8607
G-mean	0.9158	0.9534	0.8496

### 3.4. Separating Results of the Entire Study Site

The results in Sections 3.2 and 3.3 indicated the satisfactory performance of the XGBoost model in filtering salt-marsh vegetation. Therefore, the trained XGBoost model was applied to the entire study site (Figure 7a,i). A total of 25,740,656 and 217,357,732 points were identified as ground and vegetation. The ground points were sparse and only accounted for 11% of the total points. It can be observed that the various types of salt-marsh vegetation points were effectively separated, while the ground points in the bare mudflat areas were completely preserved. The distribution and orientation of the tidal creeks were easily discernible. Additionally, it can be observed from Figure 7i that the acquired ground points are denser at the nadir regions, i.e., strip effects existed in the ground points. The strip effects were due to the normal incidence at nadir regions, making a better penetration performance for the lasers within the dense vegetation.



**Figure 7.** Separating results of the entire study site. (a–h): Ground and nonground/vegetation points acquired by XGBoost, AdaBoost, RF, Catboost, PMF, SF, CSF, and, CIF, respectively. (i–p): Ground points acquired by XGBoost, AdaBoost, RF, Catboost, PMF, SF, CSF, and CIF, respectively. The ground/vegetation points after separating are 25,740,656/217,357,732, 25,757,205/217,341,183, 24,983,927/218,114,461, 25,675,001/217,423,387, 51,229,362/191,869,026, 9,8575,115/144,523,273, 141,987,475/101,110,913, and 130,563,383/112,535,005 for XGBoost, AdaBoost, RF, Catboost, PMF, SF, CSF, and CIF, respectively.

#### 4. Discussion

We compared the AUC and G-mean values of the proposed method with those of other commonly used methods to the validation set to further demonstrate superiority (Figure 8 and Table 3). These methods included three machine-learning methods (i.e., RF, AdaBoost, and Categorical Boosting (CatBoost)), three advanced filtering methods based on the geometric data (i.e., PMF, SF, and CSF), and a filtering method based on the Phong

model corrected intensity (i.e., CIF) [24]. To ensure a fair comparison, the features selected and the training and test datasets used for RF, AdaBoost, and CatBoost were consistent with that of the XGBoost; the parameters of RF, AdaBoost, and CatBoost were tuned to achieve the best results. Obviously, PMF, SF, CSF, and CIF all had a considerable misclassification rate. By contrast, the four machine-learning methods achieved higher AUC and G-mean values. This is because the common point-cloud filtering methods rely purely on either geometric or intensity information of the point cloud, which makes it difficult to effectively separate the various types of dense vegetation from the ground. By integrating both the geometric and intensity data as input features, machine-learning methods autonomously learn the classification patterns of vegetation and ground and can achieve an accurate separation of ground and vegetation points. XGBoost achieved higher AUC and G-mean values compared to RF, AdaBoost, and CatBoost in Regions 1 and 3. The three machine-learning methods performed similarly in Region 2, with RF having marginally higher AUC and G-mean values. In general, XGBoost is more efficient in separating salt-marsh vegetation point clouds than the other three machine-learning methods. The application of RF, AdaBoost, CatBoost, PMF, SF, CSF, and CIF to the entire study site is shown in Figure 7. A considerable number of vegetation points were misclassified as ground points for PMF, SF, CSF, and CIF. No obvious differences were observed in the separating results for XGBoost, RF, AdaBoost, and CatBoost.

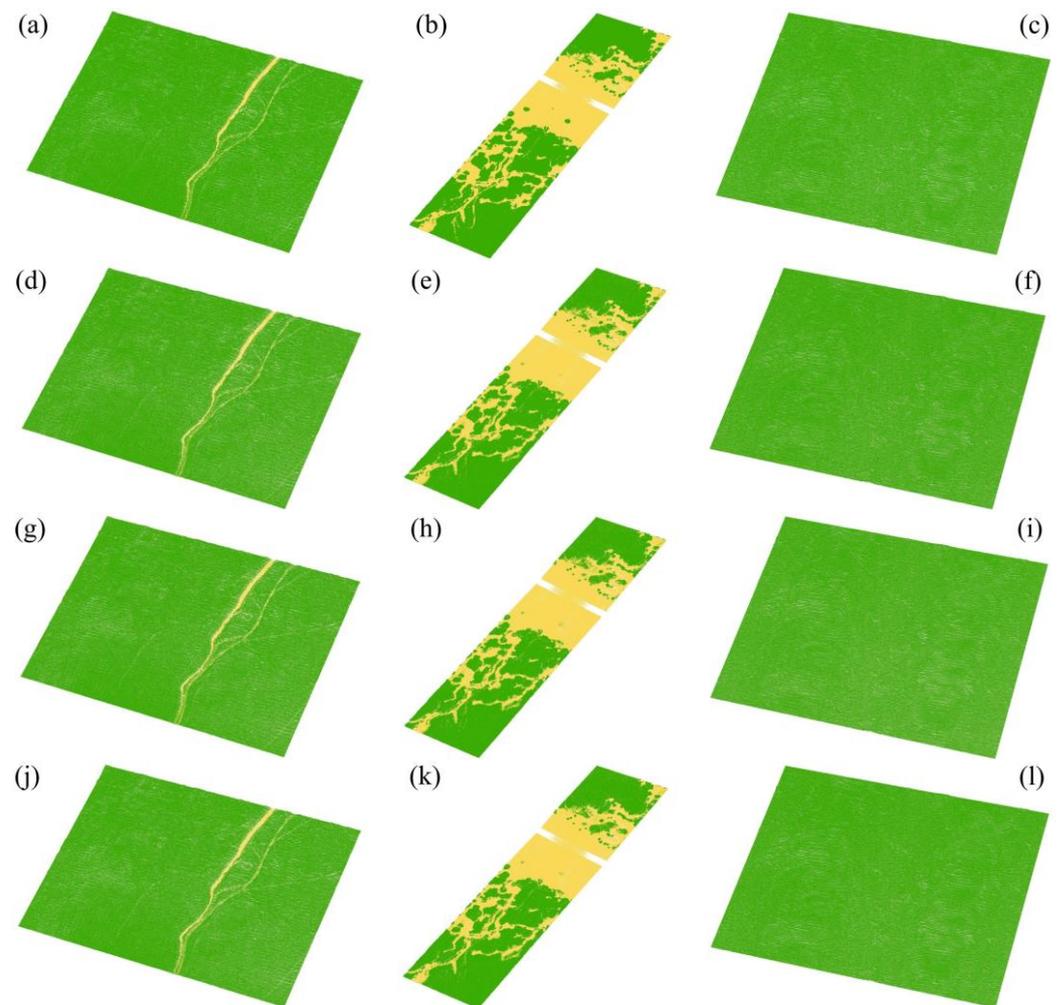
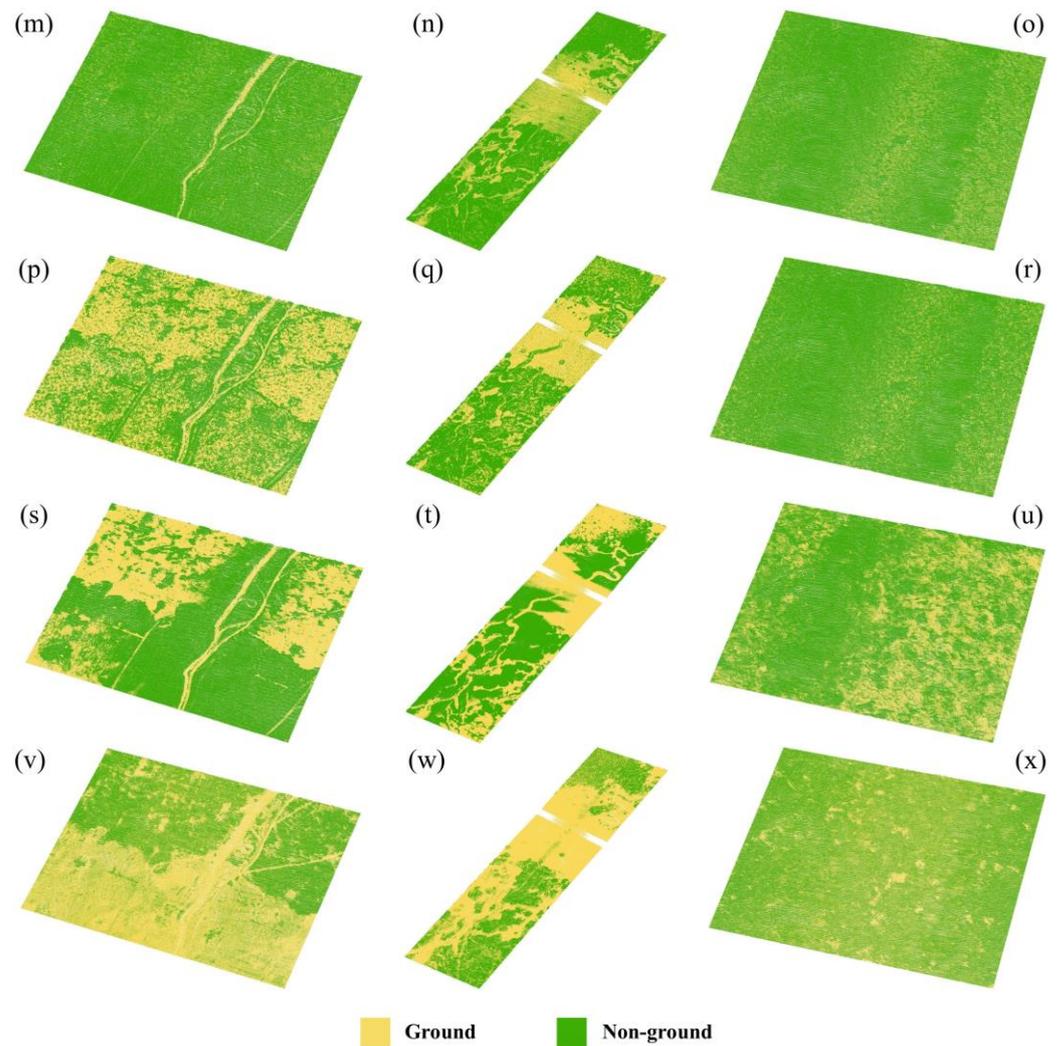


Figure 8. Cont.



**Figure 8.** Separating results comparison of different point-cloud filtering methods on the validation set. (a–c) Manual. (d–f) RF. (g–i) AdaBoost. (j–l) CatBoost. (m–o) PMF. (p–r) SF. (s–u) CSF. (v–x) CIF.

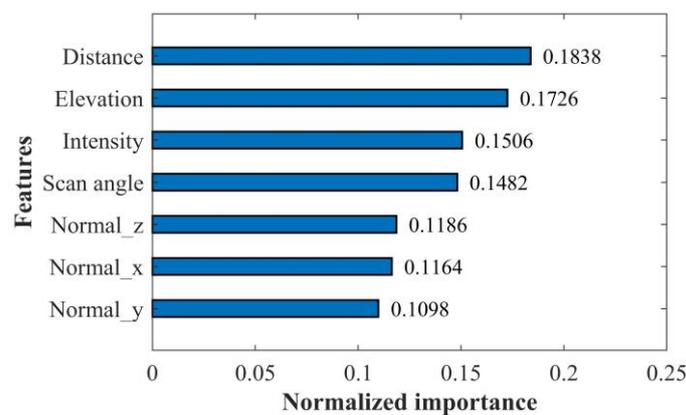
**Table 3.** Quantitative evaluation results of the validation set for different filtering methods.

	Region 4		Region 5		Region 6	
	AUC	G-Mean	AUC	G-Mean	AUC	G-Mean
XGBoost	0.9191	0.9158	0.9535	0.9534	0.8607	0.8496
RF	0.9069	0.9023	0.9551	0.9550	0.8554	0.8433
AdaBoost	0.9032	0.8984	0.9516	0.9515	0.8124	0.7911
CatBoost	0.9136	0.9098	0.9549	0.9548	0.8363	0.8204
PMF	0.7350	0.7226	0.8019	0.7986	0.7232	0.7157
SF	0.6495	0.6169	0.7301	0.7296	0.7017	0.6829
CSF	0.7366	0.7251	0.6820	0.6819	0.5627	0.5588
CIF	0.6876	0.6237	0.7639	0.7361	0.5209	0.5092

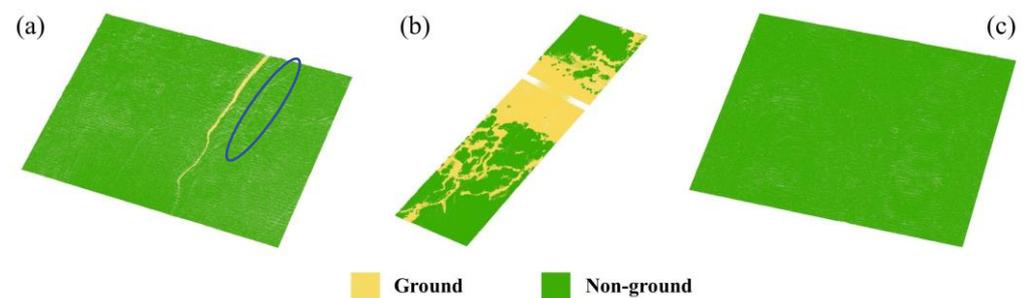
Additionally, the training time for the three machine-learning methods was recorded and compared to evaluate the efficiency. Experiments were conducted on a desktop (32-GB RAM and Inter Core i7-11700K CPU at 3.6 GHz) using Python 3.8.3 and Scikit-learn 1.0.1. The training times were 0.11, 0.74, 14.22, and 0.27 h for RF, XGBoost, AdaBoost, and CatBoost, respectively. RF had the shortest training time because each decision tree is independent, which allows for parallel computation to accelerate the model training. Additionally, feature and sample subsampling in RF can decrease the computational work. Though XGBoost can also be parallelized, it requires more computational resources when

handling large datasets and high-dimensional features, leading to a slower computational speed. AdaBoost had the longest training time because it updates sample weights in each iteration, which involves resampling and significantly increases the training time. CatBoost had the second shortest training time after RF. This is because CatBoost uses a symmetric decision-tree structure, which can speed up training. In summary, both RF and CatBoost have shorter training times, but they slightly underperform compared to XGBoost. XGBoost achieves the highest overall accuracy, but its training time is slightly longer than that of RF and CatBoost. AdaBoost needs the longest training time compared to RF, XGBoost, and CatBoost, and its overall accuracy is the lowest.

The feature importance for each variable in the XGBoost model was calculated (Figure 9). The importance ranks highest for distance, followed by elevation, intensity, and scan angle, with the importance of normal vectors being relatively lower. This indicates that distance and scan angle play significant roles in the XGBoost model. To further analyze the importance of distance and scan-angle features in compensating for the intensity data, we constructed a new XGBoost model after removing the scan angle and distance features. The separating results for Regions 4, 5, and 6 are shown in Figure 10. Obviously, the tidal creek was misclassified as nonground points in Region 4 (blue ellipse in Figure 10a), and there were almost no ground points in Region 6. The AUC and G-mean values of the XGBoost model for Regions 4, 5, and 6 were 0.6845 and 0.6075, 0.9512 and 0.9501, and 0.5013 and 0.0516, respectively. Compared to the XGBoost model without excluding the scan angle and distance features, the accuracy of Regions 4 and 6 decreased substantially, while the accuracy of Region 5 marginally declined. This indicates that the inclusion of distance and scan-angle features can effectively eliminate their effects on intensity data, achieving a better separation between ground and nonground points, especially in regions with dense vegetation coverage.



**Figure 9.** Order of feature importance of the training set. Normal\_x, Normal\_y, and Normal\_z represent the x, y, and z components of the normal vectors, respectively.



**Figure 10.** Separating results of XGBoost model after excluding scan angle and distance features. (a) Region 4. (b) Region 5. (c) Region 6.

In the practical applications of the proposed method, the distance, scan angle, elevation, normal vectors, and original intensity features of all point clouds are initially calculated and acquired. Subsequently, several typical regions (e.g., regions with different vegetation species or topography features) are selected and accurately classified into ground and nonground points manually or using traditional point-cloud filtering methods. These selected regions are then divided into training and test sets, and the XGBoost model is constructed with hyperparameter optimization. Ultimately, all point clouds can be precisely separated into ground and nonground points using the trained XGBoost model. Based on the constructed model, the combination of the selected features can accurately and intelligently predict the category of each point. Only the original user-provided data and the corresponding fundamental derived quantities are needed.

The proposed instantaneous scan geometry estimation method for a drone multiline LiDAR system is based on the linear or near-linear alignment of the scanning points (Figure 3), which is applicable to different LiDAR systems with the scanning modes (e.g., swing scanning and rotating regular polyhedron scanning) that satisfy this point alignment. For other scanning modes (e.g., rotating elliptical cylindrical scanning), the scanning points are nonlinear alignment. The proposed instantaneous scan geometry estimation method is infeasible in this circumstance. However, the proposed filtering method remains applicable as long as the scan angles are recorded by the LiDAR system.

This study focused on regions with various salt-marsh vegetation types and topography characteristics along the coastal zones for the training, testing, and validation of the XGBoost model. The results consistently demonstrated enhanced applicability and versatility. However, the coastal zones have extremely complex ecological environments, with variations in salt-marsh vegetation types across different regions (e.g., *Suaeda glauca* and *Tamarix chinensis* in the Yellow River Estuary in China) and diverse sedimentary geomorphological landforms in different estuarine deltas. Further analyses are needed in future studies to assess the filtering performance of the proposed method under varying environmental conditions in different coastal zones. Moreover, the proposed method can be further explored for different coastal types (e.g., sandy and rocky coasts) and vegetation species (e.g., mangroves).

The proposed method provides a new alternative mode for massive point-cloud filtering, as well as a novel approach for LiDAR-intensity correction, and a new strategy for the combination of the LiDAR geometric and radiometric data. Additionally, this method is not limited to a UAV-LiDAR system and can be extended to various platforms, including terrestrial and mobile LiDAR systems. Meanwhile, this method can be applied to various scenarios, e.g., urban and forest. For example, in steep and dense vegetation coverage forest regions where the ground points on the slope may be at the same elevation as the vegetation points, traditional filtering methods can hardly achieve satisfactory results. The proposed method can be used as an alternative solution in such scenarios.

## 5. Conclusions

In this study, we accurately estimate the instantaneous scan geometry based on the scanning principles of the drone multiline LiDAR systems and propose a new alternative point-cloud filtering method based on XGBoost for coastal salt-marsh vegetation. The method can substitute the complex intensity correction that is typically required for intensity-based filtering methods. No cumbersome parameterization, dimensionality reduction/projection, and prior understanding of the specific scene/data are needed for the proposed model. The scan angle, distance, intensity, elevation, and normal vectors are selected as input features to build a point-cloud filtering model based on XGBoost, which can realize the combination of geometric and intensity data for filtering. The results indicate that drone multiline LiDAR is a very promising technology for salt-marsh terrain measuring and the proposed method exhibits superiority in the acquisition of the ground points in terms of accuracy, efficiency, and robustness. However, only the fundamental quantities are empirically selected as input features. In future work, more additional fea-

tures and quantities can be considered and applied to more complex coastal environments to further improve the generalizability and accuracy of the proposed method. Additionally, the strategy of using the basic geometric and radiometric metrics with machine-learning models provides a new solution for the intelligent interpretation of LiDAR data under different modalities and platforms.

**Author Contributions:** X.W.: Conceptualization, Formal analysis, Writing—original draft. K.T.: Writing—review and editing, funding acquisition, Supervision. S.L.: Conceptualization, Methodology, Formal analysis, Writing—review and editing. F.W.: Resources, Validation, Software. P.T.: Supervision. Y.W.: Validation. X.C.: Resources. All authors have read and agreed to the published version of the manuscript.

**Funding:** The study was supported by the science and technology funds from the Guangzhou Urban Planning and Design Survey Research Institute (Grant RDI2220201008), National Natural Science Foundation of China (Grant 42171425, Grant 41901399, Grant 42004158), Chongqing Municipal Bureau of Science and Technology (Grant CSTB2022NSCQ-MSX1254), Science and Technology Commission of Shanghai Municipality (Grant 20DZ1204700, Grant 22ZR1420900), and Hunan Provincial Key Laboratory of Geo-Information Engineering in Surveying, Mapping and Remote Sensing, Hunan University of Science and Technology (Grant E22335).

**Data Availability Statement:** Data will be made available upon request.

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

## References

- Noujas, V.; Thomas, K.V.; Badarees, K.O. Shoreline management plan for a mudbank dominated coast. *Ocean Eng.* **2016**, *112*, 47–65. [[CrossRef](#)]
- Yan, J.F.; Zhao, S.Y.; Su, F.Z.; Du, J.X.; Feng, P.F.; Zhang, S.X. Tidal flat extraction and change analysis based on the RF-W model: A case study of Jiaozhou Bay, East China. *Remote Sens.* **2021**, *13*, 1436. [[CrossRef](#)]
- Macreadie, P.I.; Anton, A.; Raven, J.A.; Beaumont, N.; Connolly, R.M.; Friess, D.A.; Kelleway, J.J.; Kennedy, H.; Kuwae, T.; Lavery, P.S.; et al. The future of Blue Carbon science. *Nat. Commun.* **2019**, *10*, 13. [[CrossRef](#)] [[PubMed](#)]
- Wang, F.M.; Sanders, C.J.; Santos, I.R.; Tang, J.W.; Schuerch, M.; Kirwan, M.L.; Kopp, R.E.; Zhu, K.; Li, X.Z.; Yuan, J.C.; et al. Global blue carbon accumulation in tidal wetlands increases with climate change. *Nat. Sci. Rev.* **2021**, *8*, 11. [[CrossRef](#)] [[PubMed](#)]
- Rolando, J.L.; Hodges, M.; Garcia, K.D.; Krueger, G.; Williams, N.; Carr, J.C.; Robinson, J.; George, A.; Morris, J.; Kostka, J.E. Restoration and resilience to sea level rise of a salt marsh affected by dieback events. *Ecosphere* **2023**, *14*, 16. [[CrossRef](#)]
- Paul, M.; Bischoff, C.; Koop-Jakobsen, K. Biomechanical traits of salt marsh vegetation are insensitive to future climate scenarios. *Sci. Rep.* **2022**, *12*, 21272. [[CrossRef](#)] [[PubMed](#)]
- Kulawardhana, R.W.; Popescu, S.C.; Feagin, R.A. Fusion of lidar and multispectral data to quantify salt marsh carbon stocks. *Remote Sens. Environ.* **2014**, *154*, 345–357. [[CrossRef](#)]
- White, S.M.; Madsen, E.A. Tracking tidal inundation in a coastal salt marsh with Helikite airphotos: Influence of hydrology on ecological zonation at Crab Haul Creek, South Carolina. *Remote Sens. Environ.* **2016**, *184*, 605–614. [[CrossRef](#)]
- Tang, Y.N.; Ma, J.; Xu, J.X.; Wu, W.B.; Wang, Y.C.; Guo, H.Q. Assessing the impacts of tidal creeks on the spatial patterns of coastal salt marsh vegetation and its aboveground biomass. *Remote Sens.* **2022**, *14*, 1839. [[CrossRef](#)]
- Jin, C.; Gong, Z.; Shi, L.; Zhao, K.; Tinoco, R.O.; San Juan, J.E.; Geng, L.; Coco, G. Medium-term observations of salt marsh morphodynamics. *Front. Mar. Sci.* **2022**, *9*, 13. [[CrossRef](#)]
- Kang, Y.; Ding, X.; Xu, F.; Zhang, C.; Ge, X. Topographic mapping on large-scale tidal flats with an iterative approach on the waterline method. *Estuar. Coast. Shelf Sci.* **2017**, *190*, 11–22. [[CrossRef](#)]
- Gao, W.L.; Shen, F.; Tan, K.; Zhang, W.G.; Liu, Q.X.; Lam, N.S.N.; Ge, J.Z. Monitoring terrain elevation of intertidal wetlands by utilising the spatial-temporal fusion of multi-source satellite data: A case study in the Yangtze (Changjiang) Estuary. *Geomorphology* **2021**, *383*, 12. [[CrossRef](#)]
- Chen, C.; Zhang, C.; Schwarz, C.; Tian, B.; Jiang, W.; Wu, W.; Garg, R.; Garg, P.; Aleksandr, C.; Mikhail, S.; et al. Mapping three-dimensional morphological characteristics of tidal salt-marsh channels using UAV structure-from-motion photogrammetry. *Geomorphology* **2022**, *407*, 108235. [[CrossRef](#)]
- Yang, B.; Ali, F.; Zhou, B.; Li, S.; Yu, Y.; Yang, T.; Liu, X.; Liang, Z.; Zhang, K. A novel approach of efficient 3D reconstruction for real scene using unmanned aerial vehicle oblique photogrammetry with five cameras. *Comput. Electr. Eng.* **2022**, *99*, 107804. [[CrossRef](#)]
- Taddia, Y.; Pellegrinelli, A.; Corbau, C.; Franchi, G.; Staver, L.W.; Stevenson, J.C.; Nardin, W. High-resolution monitoring of tidal systems using UAV: A case study on poplar island, MD (USA). *Remote Sens.* **2021**, *13*, 1364. [[CrossRef](#)]

16. Wang, X.X.; Xiao, X.M.; Zou, Z.H.; Chen, B.Q.; Ma, J.; Dong, J.W.; Doughty, R.B.; Zhong, Q.Y.; Qin, Y.W.; Dai, S.Q.; et al. Tracking annual changes of coastal tidal flats in China during 1986–2016 through analyses of Landsat images with Google Earth Engine. *Remote Sens. Environ.* **2020**, *238*, 15. [[CrossRef](#)]
17. Xu, N.; Ma, Y.; Yang, J.; Wang, X.H.; Wang, Y.J.; Xu, R. Deriving tidal flat topography using ICESat-2 laser altimetry and Sentinel-2 imagery. *Geophys. Res. Lett.* **2022**, *49*, 10. [[CrossRef](#)]
18. Chen, C.P.; Zhang, C.; Tian, B.; Wu, W.T.; Zhou, Y.X. Tide2Topo: A new method for mapping intertidal topography accurately in complex estuaries and bays with time-series Sentinel-2 images. *ISPRS-J. Photogramm. Remote Sens.* **2023**, *200*, 55–72. [[CrossRef](#)]
19. Brunetta, R.; Duo, E.; Ciavola, P. Evaluating short-term tidal flat evolution through UAV surveys: A case study in the Po Delta (Italy). *Remote Sens.* **2021**, *13*, 2322. [[CrossRef](#)]
20. Xie, W.M.; Guo, L.C.; Wang, X.Y.; He, Q.; Dou, S.T.; Yu, X. Detection of seasonal changes in vegetation and morphology on coastal salt marshes using terrestrial laser scanning. *Geomorphology* **2021**, *380*, 10. [[CrossRef](#)]
21. Zhou, W.; Chen, F.L.; Guo, H.D.; Hu, M.Y.; Li, Q.; Tang, P.P.; Zheng, W.W.; Liu, J.A.; Luo, R.P.; Yan, K.K.; et al. UAV Laser scanning technology: A potential cost-effective tool for micro-topography detection over wooded areas for archaeological prospection. *Int. J. Digit. Earth* **2020**, *13*, 1279–1301. [[CrossRef](#)]
22. Kim, H.; Kim, Y.; Lee, J. Tidal creek extraction from airborne LiDAR data using ground filtering techniques. *KSCE J. Civ. Eng.* **2020**, *24*, 2767–2783. [[CrossRef](#)]
23. Sun, X.B.; Wang, S.K.; Liu, M. A novel coding architecture for multi-line LiDAR point clouds based on clustering and convolutional LSTM network. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 2190–2201. [[CrossRef](#)]
24. Tao, P.J.; Tan, K.; Ke, T.; Liu, S.; Zhang, W.G.; Yang, J.R.; Zhu, X.J. Recognition of ecological vegetation fairy circles in intertidal salt marshes from UAV LiDAR point clouds. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *114*, 9. [[CrossRef](#)]
25. Li, H.F.; Ye, C.M.; Guo, Z.X.; Wei, R.L.; Wang, L.X.; Li, J. A fast progressive TIN densification filtering algorithm for airborne LiDAR data using adjacent surface information. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2021**, *14*, 12492–12503. [[CrossRef](#)]
26. Zhang, K.Q.; Chen, S.C.; Whitman, D.; Shyu, M.L.; Yan, J.H.; Zhang, C.C. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 872–882. [[CrossRef](#)]
27. Vosselman, G. Slope based filtering of laser altimetry data. *Int. Arch. Photogramm. Remote Sens.* **2000**, *33*, 935–942.
28. Zhang, W.M.; Qi, J.B.; Wan, P.; Wang, H.T.; Xie, D.H.; Wang, X.Y.; Yan, G.J. An easy-to-use airborne LiDAR data filtering method based on cloth simulation. *Remote Sens.* **2016**, *8*, 501. [[CrossRef](#)]
29. Tan, K.; Chen, J.; Qian, W.W.; Zhang, W.G.; Shen, F.; Cheng, X.J. Intensity data correction for long-range terrestrial laser scanners: A case study of target differentiation in an intertidal zone. *Remote Sens.* **2019**, *11*, 331. [[CrossRef](#)]
30. Errington, A.F.C.; Daku, B.L.F. Temperature compensation for radiometric correction of terrestrial LiDAR intensity data. *Remote Sens.* **2017**, *9*, 356. [[CrossRef](#)]
31. Tan, K.; Cheng, X.J. Correction of incidence angle and distance effects on TLS intensity data based on reference targets. *Remote Sens.* **2016**, *8*, 251. [[CrossRef](#)]
32. Sanchiz-Viel, N.; Bretagne, E.; Mouaddib, E.; Dassonville, P. Radiometric correction of laser scanning intensity data applied for terrestrial laser scanning. *ISPRS-J. Photogramm. Remote Sens.* **2021**, *172*, 1–16. [[CrossRef](#)]
33. Poullain, E.; Garestier, F.; Levoy, F.; Bretel, P. Analysis of ALS intensity behavior as a function of the incidence angle in coastal environments. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2016**, *9*, 313–325. [[CrossRef](#)]
34. Zhao, J.H.; Chen, M.Y.; Zhang, H.M.; Zheng, G. A hovercraft-borne LiDAR and a comprehensive filtering method for the topographic survey of mudflats. *Remote Sens.* **2019**, *11*, 1646. [[CrossRef](#)]
35. Maxwell, A.E.; Warner, T.A.; Fang, F. Implementation of machine-learning classification in remote sensing: An applied review. *Int. J. Remote Sens.* **2018**, *39*, 2784–2817. [[CrossRef](#)]
36. Sheykhmousa, M.; Mahdianpari, M.; Ghanbari, H.; Mohammadimanesh, F.; Ghamisi, P.; Homayouni, S. Support vector machine versus random forest for remote sensing image classification: A meta-analysis and systematic review. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2020**, *13*, 6308–6325. [[CrossRef](#)]
37. Zhong, L.H.; Hu, L.N.; Zhou, H. Deep learning based multi-temporal crop classification. *Remote Sens. Environ.* **2019**, *221*, 430–443. [[CrossRef](#)]
38. Liu, S.; Tan, K.; Tao, P.J.; Yang, J.R.; Zhang, W.G.; Wang, Y.J. Rigorous density correction model for single-scan TLS point clouds. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 18. [[CrossRef](#)]
39. Chen, T.Q.; Guestrin, C.; Assoc Comp, M. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
40. Nguyen, M.H.; de la Torre, F. Optimal feature selection for support vector machines. *Pattern Recognit.* **2010**, *43*, 584–591. [[CrossRef](#)]
41. Yan, W.Y.; Shaker, A. Radiometric correction and normalization of airborne LiDAR intensity data for improving land-cover classification. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 7658–7673.
42. 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](#)]
43. Lusted, L.B. Decision-making studies in patient management. *N. Engl. J. Med.* **1971**, *284*, 416–424. [[CrossRef](#)] [[PubMed](#)]

- 
44. Bradley, A.P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognit.* **1997**, *30*, 1145–1159. [[CrossRef](#)]
  45. He, H.; Garcia, E.A. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.* **2009**, *21*, 1263–1284.

**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.