IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 13, 2019, accepted December 17, 2019, date of publication December 26, 2019,
date of current version January 7, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962385

A Physical Process Driven Digital Terrain Model Generating Method Based on D-NURBS

DANLEI YE[1,2], XIN JIANG[1,2,3], GUANYING HUO[1,2], CHENG SU[1,2], ZEHONG LU[1,4], BOLUN WANG[1,2], AND ZHIMING ZHENG[1,2]

[1]Key Laboratory of Mathematics, Informatics and Behavioral Semantics (LMIB), School of Mathematics and Systems Science, Beihang University, Beijing 100191, China
[2]Peng Cheng Laboratory, Shenzhen 518066, China
[3]Beijing Advanced Innovation Center for Big Data and Brain Computing (BDBC), Beihang University, Beijing 100191, China
[4]School of Mathematical Sciences, Peking University, Beijing 100871, China

Corresponding authors: Xin Jiang (jiangxin@buaa.edu.cn) and Guanying Huo (gyhuo@buaa.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1107402, in part by the Beijing Natural Science Foundation under Grant Z180005, and in part by the National Natural Science Foundation of China under Grant 11290141.

ABSTRACT Airborne light detection and ranging (LiDAR) technology is becoming the primary method for generating high-resolution digital terrain models (DTMs), which is essential for commercial and scientific uses. In order to generate DTMs, non-ground features as buildings, vehicles, and vegetation must be recognized and distinguished from the LiDAR point cloud. However, various degrees of errors may accumulate in the separated filtering and modeling processes. In this paper, a novel physical process driven DTM generating method for airborne LiDAR measurement is proposed, which combines the point cloud classification and surface fitting process simultaneously. Actually, the physical dynamic process is integrated with the common non-uniform rational b-splines (NURBS) model under the corresponding parameter mediation. The experimental results show that the proposed method is efficacious in reducing errors and have a nice performance in terrain fitting.

INDEX TERMS Digital terrain model, physical process driven fitting, NURBS, LiDAR point cloud.

I. INTRODUCTION

High-resolution digital terrain models (DTMs) are pivotal for commercial and scientific uses such as urban planning, path management, forest administration, flood simulation, and debris monitoring [1]–[4]. To provide sufficient data resource for DTMs, light detection and ranging (LiDAR) technology is widely used to scan the designated areas and obtain point cloud [5]–[7].

In order to generate superior DTMs, one crucial step is to divide LiDAR points into ground and non-ground ones, which is also called the filtering process. During the past decades, numerous filtering algorithms have been proposed to extract ground points from LiDAR point cloud. These methods can be mainly categorized as slope-based methods, mathematical morphology-based methods, and surface-based methods [8].

The associate editor coordinating the review of this manuscript and approving it for publication was Giambattista Gruosso.

In the slope-based methods, it is assumed that the slope variation varies greatly between different types of point clouds. The slope change of the ground in the neighborhood is usually gradual, while the slope change between the buildings and the ground is relatively large [9], [10]. Sithole and Vosselman [11] proposed a slope-based filter to identify ground data by comparing the slope between LiDAR points and their neighbors. If the maximum slope between the selected point and any point within a given circle is less than a predefined threshold, the point is classified as a ground measurement. In [12], the slope of adjacent points on the scan line along some specified direction has been considered. In order to improve the computational efficiency, [13] extended the single direction method to multi-direction scan line. Generally, slope-based filtering algorithm is suitable for the situation where the difference between the terrain slope and the non-ground object slope is significant, for instance, the urban terrain modeling.

Another typical approach is the mathematical morphology-based method, which converts LiDAR points into regular

gray-scale grid images through elevation, and then identify the shapes of buildings, cars and trees according to the changes in gray-scale [14]–[16]. Zhang *et al.* [18] present two different approaches for the filtering of laser-scanner data, namely the use of linear prediction and the use of dual rank filters. Specifically, the dual rank filtering algorithm performs a remarkable filtering effect in filtering artificial objects such as buildings. However, it needs interactive control and some prior knowledge to set the necessary parameters correctly. To improve it, Zhang *et al.* [18] develops a progressive morphological filter which can automatically extract ground points from lidar measurements with tiny user interaction.

For surface-based filtering algorithms, the ground can be gradually approached by iteratively selecting ground measurements from the original data sets. Haugerud and Harding [19] develop an algorithm to filter non-ground points by comparing the local curvature measured by points. This algorithm is proceeded by iteratively deleting tree vertices from the irregular triangular network (TIN) constructed from lidar measurement data. Axelsson [20] introduces a method to determine the ground surface and classify buildings and electrical power lines. This classification algorithm is based on the minimum description length criterion. Mongus and Žalik [21] iterates a thin plate spline interpolated surface towards the ground, while points residuals from the surface are inspected at each iteration, with a gradually decreasing window size.

The aforementioned filtering algorithms have shown successful performance in data experiments. However, various degrees of errors during may accumulate in the modeling process. In fact, on one hand, the filtering algorithm may generate potential errors in distinguishing ground points from non-ground points. On the other hand, due to the shielding effect of non-ground points above the ground, there may lack of sufficient information to describe the entire terrain which also induces modeling errors. Additionally, there are also errors in generating DTM surface based on the filtered points due to the accuracy of fitting methods. Accordingly, many methods of generating DTM surface with filtered points suffers from error accumulating from independent filtering and modeling process.

To cope with these problems, we propose a novel physical process driven DTM generating method in dealing with the LiDAR point cloud data. A physical process induced data fitting technology is introduced to integrate the filtering and modeling process. This idea is mainly motivated by the scenery that, for an initial surface among the LiDAR points, if it is attracted by the LiDAR points through some kind of force, which can be governed by a set of specialized motion equations, it will gradually change its shape and move toward the points in resulting a superior fitting. In this way, one can obtain the DTM directly from the preprocessed LiDAR point cloud, avoiding the error arising from separated filtering and modeling processes.

In this paper, we propose a holistic DTM algorithm driven by physical process, which combines the point cloud classification and surface fitting process simultaneously. Actually, we integrate the physical dynamic process with the common non-uniform rational b-splines (NURBS) model, and automatically generates continuous and smooth surface models under the corresponding parameter mediation. In this way, all the parameters in the modeling process do have corresponding physical meanings, which is convenient for users to understand and adjust, showing the operating advantage of the proposed method. Moreover, the applied dynamic NURBS (D-NURBS) surface, also has common analytic expression and is easy to store and transmit. Various data simulating experiments on benchmark terrain surface demonstrate that our method is efficacious in reducing errors and have a nice performance in terrain fitting.

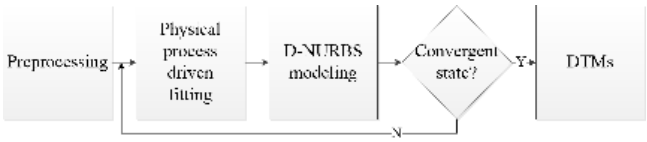

FIGURE 1. Framework of the physical process driven DTM generating method based on D-NURBS.

The arrangement of this paper is as follows. Section II introduces the mechanism of the fitting technology driven by physical process and its integration with selected geometry modeling methods. In Section III, experimental results are presented. Conclusions are made in Section IV.

II. METHOD

The establishment process of physical process driven DTM could be demonstrated as four steps as follows: To begin with, we preprocess the obtained LiDAR point cloud data to remove noise points; Secondly, a specific physical process is introduced, aiming at inducing a dynamic surface to fit the preprocessed LiDAR data.; In the third step, the previously designed dynamic fitting surface is characterized by a D-NUBRS model. At last, the embedded physical process in D-NUBRS model is iterated until it evolves into a static equilibrium state, which indicates the iterative calculation has reached the convergence state. At this point, the algorithm is terminated and the final DTM is obtained. The whole process is shown in Fig. 1.

A. PREPROCESS

Actually, the original LiDAR points data set obtained is usually scattered and irregular. Thus, the first stage of our research comprises getting well distributed data points.

Due to the influence of hardware equipment and environment, the collected data points are mixed with points that are neither ground points nor objects on the ground. Since these noise or outlier points largely affect the generation of DTM, the preprocessing of the initial data set becomes necessary and critical.

In this research we introduce a statistical filtering technique to remove noise points from the LiDAR point cloud. This method is based on the statistical calculation of the

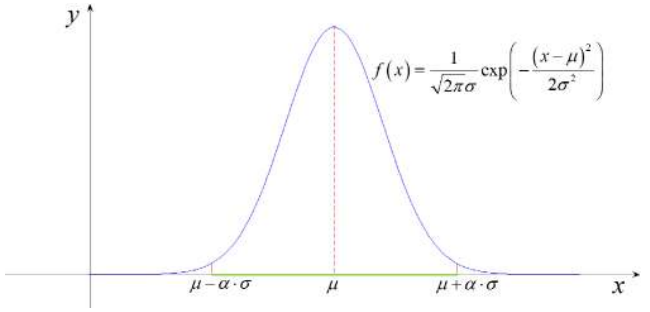

FIGURE 2. Gaussian distribution of the averaged point-pair distance. The x-axis represents the average distance from one point to all its neighbors, and the y-axis represents the number of points.

neighborhood distance distribution [22]. Concretely, for each data point i, the average distance D_i between the target point and its neighbors is obtained by calculating the mean value of point pairs' Euclidean distance.

$$D_i = \frac{1}{k}\sum_{j\in N(i)} |p_j - p_i| \tag{1}$$

Here k is a preset number.$N(i)$ is the set of k neighbors.p_i is the generalized coordinates of point i.

We then consider the distribution of D_i, which is supposed to follow a Gaussion distribution, as shown in Fig. 2. Statistically speaking, the shape of the distance distribution depends on the mean value μ and standard deviation σ, which provides a criterion for identifying outlier points. Literally, a point whose corresponding average distance is beyond the standard range $\mu \pm \alpha \cdot \sigma$ may be identified as an outlier and removed from the point set. Here the coefficient α is an adjustable parameter depending on the actual experimental data. In this research, we set $k = 30$ and $\alpha = 2.5$. We demonstrate the preprocessing procedure in Fig.2., as shown in the picture, points located in the range of $x < \mu - \alpha \cdot \sigma$ or $x > \mu + \alpha \cdot \sigma$ are identified as outliers, while points distributed in the range of $\mu + \alpha \cdot \sigma < x < \mu + \alpha \cdot \sigma$(marked green) are preserved.

After preprocessing, the processed set of points will be utilized to build the DTM. We should mention here that points that are filtered out are not taken into account for error calculation in the following steps.

B. PHYSICAL PROCESS DRIVEN FITTING

A physical process driven model integrates mass distribution, internal deformation energy and other physical quantities into general surface geometric substrates [23]. Accordingly, this kind of model can be used to simulate the dynamic changing process of a surface, as well as characterize the process of dynamic fitting and the physical state at any moment. In this research, the proposed deformable model is governed by the laws of mechanics of continuums whose shapes can change over time. These laws, described by a set of dynamic differential equations, unify the descriptions of shape and motion.

The proposed dynamic surface fitting process is mainly controlled by a pre-defined physical process. At the very beginning, an initial surface S is supposed to be set among the

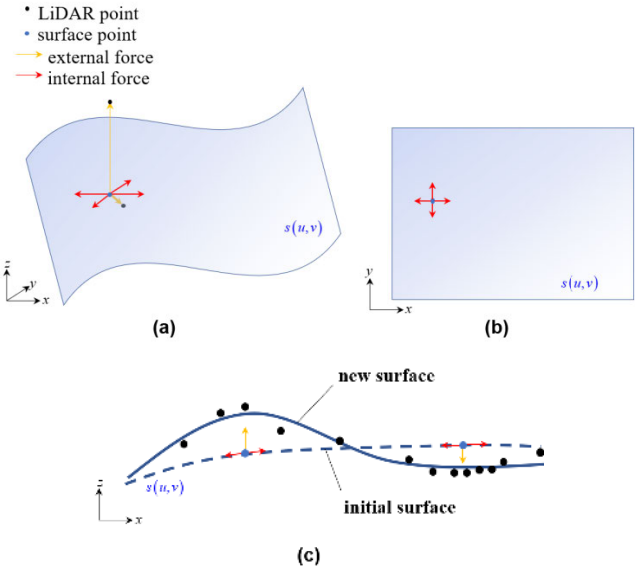

FIGURE 3. Dynamic surface fitting process. (a) The three-dimensional view. (b) XY plane view. (c) The section along the y direction.

LiDAR points, which may not fit the points properly. Assuming that each single LiDAR point performs an attractive force to a generalized point i associated with S. Here associated means i might be a point on S if S is a mesh or a control point of S if S is a spline surface. In this way, S becomes deformable and flexible since it is attracted by the LiDAR points and gradually fits the points cloud. The attracted force is considered as an external factor.

Besides, consider that S is not sufficiently soft, it has a stiffness which keeps S as flat as possible. In addition, kinetic energy, inertia and dissipated energy are also considered as internal factors in this physical process.

The overview of the proposed algorithm is illustrated in Fig. 3. Fig. 3(a) and Fig. 3(b) shows the external and internal forces acting on the surface respectively. In Fig. 3(c), the fitting surface changes from the initial shape to the new shape under the action of external forces.

Subsequently, to make the aforementioned physical system become an equilibrium one, forces induced by internal factors should be balanced with the external one, which can be described by a Lagrangian equation of motion:

$$\frac{d}{dt}\cdot\frac{\partial T}{\partial \dot{p}_i} - \frac{\partial T}{\partial p_i} + \frac{\partial F}{\partial \dot{p}_i} + \frac{\partial U}{\partial p_i} = f_i. \tag{2}$$

Here $p_i(t)$ is the generalized coordinates of point i varies with time t.

$f_i(t)$ is the resultant attractive force that acts on p_i. T, U, F respectively represent kinetic energy, potential energy and Raleigh dissipation energy. N considered $p_i(t)$ forms a N-dimensional vector $\mathbf{p}$.

Indeed, once the mechanical balancing process evolves into an equilibrium state, the surface motion becomes static. Equally, the velocity of all the points associated with S tends

to be zero. In this sense, surface S is fixed, which means the kinetic energy and inertia of the surface also becomes zero.

Since the kinetic energy $T = 0$, one may solve the motion equation by introducing a damping density function $\gamma(u, v)$. Thus, the dissipation energy of the surface can be written as

$$F = \frac{1}{2}\iint \gamma \dot{\mathbf{s}}^T \dot{\mathbf{s}} du dv = \frac{1}{2}\dot{\mathbf{p}}^T \mathbf{D}\dot{\mathbf{p}}. \tag{3}$$

Here $D(\mathbf{p}) = \iint \gamma \mathbf{J}^T \mathbf{J} du dv$, and $\mathbf{J}$ is the Jacobian matrix of the surface S.

Similarly, we define $\alpha_{ij}(u, v)$, $\beta_{ij}(u, v)$ as the elasticity functions in the parametric coordinate directions (u,v), then the elastic energy can be written as

$$\begin{aligned} U &= \frac{1}{2}\iint \left(\alpha_{11}\frac{\partial s^T}{\partial u}\frac{\partial s}{\partial u} + \alpha_{22}\frac{\partial s^T}{\partial v}\frac{\partial s}{\partial v} + \beta_{11}\frac{\partial^2 s^T}{\partial u^2}\frac{\partial^2 s}{\partial u^2}\right. \\ &\quad \left. + 2\beta_{12}\frac{\partial^2 s^T}{\partial u \partial v}\frac{\partial^2 s}{\partial u \partial v} + \beta_{22}\frac{\partial^2 s^T}{\partial v^2}\frac{\partial^2 s}{\partial v^2}\right) du dv \\ &= \frac{1}{2}\mathbf{p}^{\mathrm{T}}\mathbf{K}\mathbf{p}. \end{aligned} \tag{4}$$

Here

$$\begin{aligned} K(\mathbf{p}) = \iint &\left(\alpha_{11}\mathbf{J}_u^T\mathbf{J}_u + \alpha_{22}\mathbf{J}_v^T\mathbf{J}_v + \beta_{11}\mathbf{J}_{uu}^T\mathbf{J}_{uu}\right. \\ &\left. + 2\beta_{12}\mathbf{J}_{uv}^T\mathbf{J}_{uv} + \beta_{22}\mathbf{J}_{vv}^T\mathbf{J}_{vv}\right) du dv. \end{aligned}$$

As $f_i(t)$ is the resultant attractive force that acts on p_i, we assemble f_i into the vector $\mathbf{f}_p$ as

$$\mathbf{f}_p(\mathbf{p}) = \iint \mathbf{J}^{\mathrm{T}} f(u, v, t)\, du dv. \tag{5}$$

Here

$$\begin{aligned} \mathrm{f}(u, v, t) &= \iint \mathrm{c}(d_0 - \mathrm{s}(u, v, t))\, \delta(u - u_0, v - v_0)\, du dv \\ &= \begin{cases} \mathrm{c}(d_0 - \mathrm{s}(u, v, t)) & (u, v) = (u_0, v_0) \\ 0 & others \end{cases}. \end{aligned} \tag{6}$$

In this way, (2) can be transformed into

$$\mathbf{D}\dot{\mathbf{p}} + \mathbf{K}\mathbf{p} = \mathbf{f}_p. \tag{7}$$

(7) is the simplified Lagrangian equation of motion that is also the control condition of our dynamic model. In practical calculation, we adopt its discrete form as

$$(\mathbf{D} + \Delta t\mathbf{K})\mathbf{p}^{(t+\Delta t)} = \Delta t\mathbf{f}_p + \mathbf{D}\mathbf{p}^{(t)}. \tag{8}$$

Then we can plug in general surface representation. We can adjust the parameters in this control condition to achieve the purpose of adjusting the surface, so as to provide more suitable DTMs for different terrain conditions.

(8) is convergent, so $\mathbf{p}$ also tend to a stable equilibrium state in the iterative process. The final convergence state determines the fitting surface, that is, the DTM we need.

C. TERRAIN MODELING BASED ON D-NURBS

In this section, we mainly discuss the proper mathematical description of surface S. Generally, the non-uniform rational b-spline surface (NURBS) is a preferred to express the surface.

Basically, a NURBS form can be defined as

$$\mathbf{s}(u, v) = \frac{\sum_{i=0}^{m}\sum_{j=0}^{n}\mathbf{p}_{i,j}\omega_{i,j}B_{i,k}(u)B_{j,l}(v)}{\sum_{i=0}^{m}\sum_{j=0}^{n}\omega_{i,j}B_{i,k}(u)B_{j,l}(v)}. \tag{9}$$

Here $\mathbf{p}_{i,j}$ is the control points,$\omega_{i,j}$ is the weight coefficient and $B_{i,k}(u)$, $B_{j,l}(v)$ are the spline basis functions on u and v. In this research, since all the associated points on S are evolving with time, governing by the dynamic equation (7), the dynamic NURBS (D-NURBS) method is a beneficial choice which could characterize the time-evolving feature of the surface. Actually, for D-NURBS, the control points and weight coefficients are independent functions of time t, thus a D-NURBS surface can be expressed as

$$\mathbf{s}(u, v, t) = \frac{\sum_{i=0}^{m}\sum_{j=0}^{n}\mathbf{p}_{i,j}(t)\omega_{i,j}(t)B_{i,k}(u)B_{j,l}(v)}{\sum_{i=0}^{m}\sum_{j=0}^{n}\omega_{i,j}(t)B_{i,k}(u)B_{j,l}(v)}. \tag{10}$$

To simplify, if the weight coefficients are all 1, the D-NURBS surface degenerates into a dynamic B-spline surface., this form can greatly reduce the complexity of coefficient setting. Thus, without loss of generality, here we employ the dynamic B-spline form., In order to preserve the same fitting precision, the B-spline grid needs to be encrypted. Concretely, by applying the dynamic B-spline, dimension of the variables in (8) decreases from $3 \times 4(n+1)(m+1)$ to $3 \times (n+1)(m+1)$.By separating the (x,y,z) coordinates, the system can be further decomposed into three $(n+1)(m+1)$ dimensional systems.

$$\begin{cases} (\tilde{\mathbf{D}} + \Delta t\tilde{\mathbf{K}})\mathbf{x}^{(t+\Delta t)} = \tilde{\mathbf{D}}\mathbf{x}^{(t)} \\ (\tilde{\mathbf{D}} + \Delta t\tilde{\mathbf{K}})\mathbf{y}^{(t+\Delta t)} = \tilde{\mathbf{D}}\mathbf{y}^{(t)} \\ (\tilde{\mathbf{D}} + \Delta t\tilde{\mathbf{K}})\mathbf{z}^{(t+\Delta t)} = \Delta i\tilde{\mathbf{f}}_p + \tilde{\mathbf{D}}\mathbf{z}^{(t)} \end{cases} \tag{11}$$

In (11), $\tilde{\mathbf{D}}$, $\tilde{\mathbf{K}}$, $\tilde{\mathbf{f}}_p$ can be numerically calculated by converting the integral into a polynomial using Gaussian integration. The detailed form can be expressed as follows through the Gauss-Legendre quadrature method:

$$\begin{aligned} \tilde{\mathbf{D}}_k &= \frac{\gamma}{n}(u_{\max} - u_{\min})(v_{\max} - v_{\min}) \\ &\quad \sum_{i=0}^{n}\sum_{j=0}^{n} A_i A_j \left(N_{k_1,k_2}(u_i, v_j)\right)^2 \\ \tilde{\mathbf{K}}_k &= \frac{(u_{\max} - u_{\min})(v_{\max} - v_{\min})}{n} \\ &\quad \sum_{i=0}^{n}\sum_{j=0}^{n} A_i A_j \left[\alpha_{11}\left(\frac{\partial N_{k_1,k_2}}{\partial u}\right)^2\right. \\ &\quad + \alpha_{22}\left(\frac{\partial N_{k_1,k_2}}{\partial v}\right)^2 + \beta_{11}\left(\frac{\partial^2 N_{k_1,k_2}}{\partial u^2}\right)^2 + \beta_{12} \\ &\quad \left.\left(\frac{\partial^2 N_{k_1,k_2}}{\partial u \partial v}\right)^2 + \beta_{22}\left(\frac{\partial^2 N_{k_1,k_2}}{\partial v^2}\right)^2\right] \end{aligned}$$

$$\tilde{\mathbf{f}}_p = \frac{c\,(u_{\max}-u_{\min})\,(v_{\max}-v_{\min})}{n}\sum_{l=0}^{N}\Delta z_l \sum_{i=0}^{n}\sum_{j=0}^{n} A_i A_j \cdot N_{k_1,k_2}(u_i, v_i)\exp\left(-\frac{(u_i-u_l)^2+(v_j-v_l)^2}{(2\sigma)^2}\right) \quad (12)$$

in which

$$k_1 = \left\lfloor \frac{k}{m+1} \right\rfloor, k_2 = k - \left\lfloor \frac{k}{m+1} \right\rfloor \cdot (m+1) - 1$$

D. ERROR ANALYSIS

After constructing the spline fitting surface, the expected DTM is obtained. Based on the spatial relationship between the original data set and the generated DTM, the original data set can be naturally classified into ground points and non-ground points accordingly. Indeed, ground points are identified as the points below the DTM surface or above the surface within a pre-defined distance and non-ground points are those pre-defined distance above the surface. Then ground scatter model and object scatter model based on the initial data set can be extracted.

For the purpose of error analysis in terrain fitting, the generally criteria applied is whether the ground points and non-ground points can be accurately distinguished. A universal, quantifiable criterion is the error proportion of discrimination. Practically, the error proportion of a proposed method could be obtained by examining the following three types of errors in a benchmark data set.

Type I error: the proportion of ground points incorrectly identified as non-ground points;

Type II error: the proportion of non-ground points incorrectly identified as ground points;

Total error: the percentage of all misclassified points in the data set.

In addition, the explanations of data and error types are presented in Tab.1:

III. EXPERIMENT AND DISSCUSSION

The hillside case with 52,119 LiDAR scanning data points, which is a typical standard benchmark data set from the International Society for Photogrammetry and Remote Sensing (ISPRS), was tested for demonstrating the effect of the proposed method by comparing with the common NURBS fitting method.

TABLE 1. Description of data and calculation of indicators.

Initial data		The number of the initial points= a+b+c+d+e			
Participant data		a+b+c+d		Unused(noise)	e
		Classification results			
		Ground	Non-ground	Errors	%
Reference	ground	a	b	Type I	$\frac{b}{a+b}$
	Non-ground	c	d	Type II	$\frac{c}{c+d}$
Total					$\frac{b+c}{a+b+c+d}$

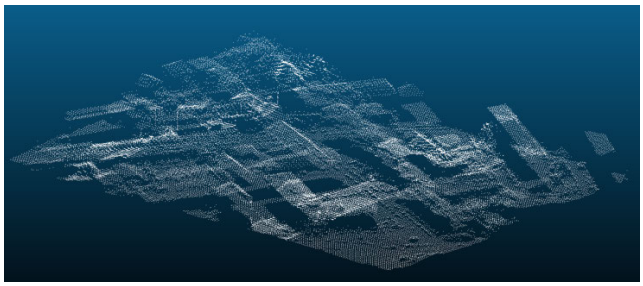

FIGURE 4. The pre-processed LiDAR points.

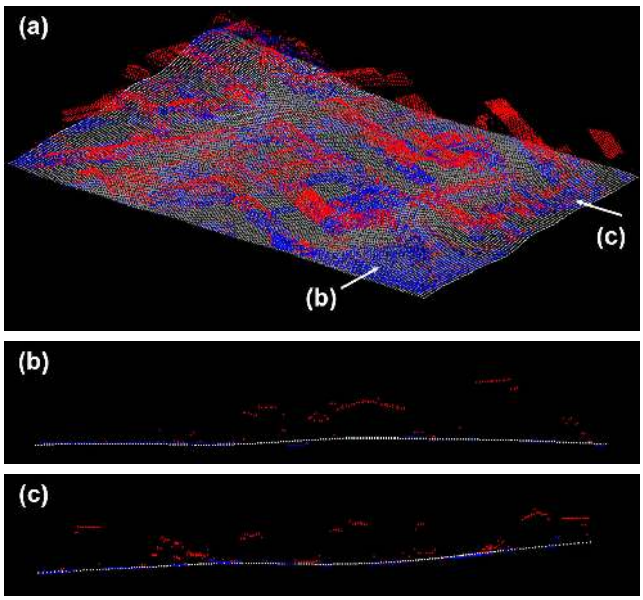

FIGURE 5. The results fitted by our proposed method. The blue and red points are marked data points. The white points are scattered points on the DTM. (a) The fitting effect of all data in 3D space. (b) The section along the x direction. (c) The section along the y direction.

These scattered and disordered LiDAR data reflect the position information of the top surface like the building roofs, but the vertical walls and the ground covered by the building are missing. In addition, the distribution of the raw LiDAR data is not even due to the scanning overlaps caused by airborne LiDAR's scanning track and angle. 51,629 LiDAR data points were preserved after preprocessing, as shown in Fig. 4.

As shown in Fig.5 and Fig.6(a), the LiDAR points has been marked and classified manually into the ground (the blue points) and the non-ground (the red points). Fig.5(a) shows the effect of the proposed method by integrating the generated DTM (white surface) and the marked LiDAR point together. In detail, two cross-sections are picked randomly to illustrate the accuracy of the proposed method, shown in Fig.5(b) and Fig.5(c). The white lines in Fig.5(b) and Fig.5(c), and the generated DTM in the corresponding cross-sections, separate the ground points and the non-ground points clearly.

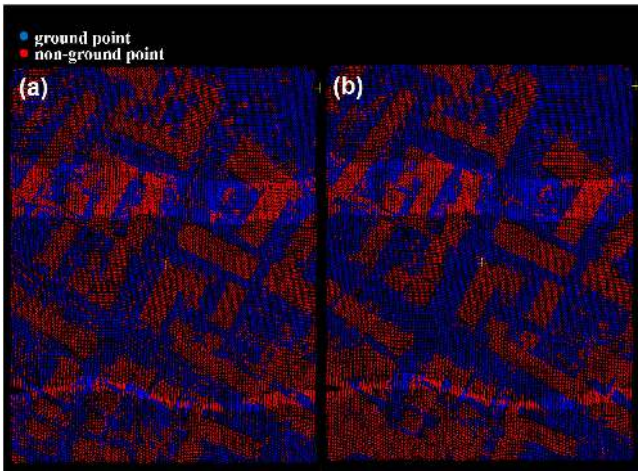

FIGURE 6. In classifications, the ground points are marked blue and the non-ground points are marked red.(a) The correct classification of data points. (b) The classification of sampling points through our proposed method.

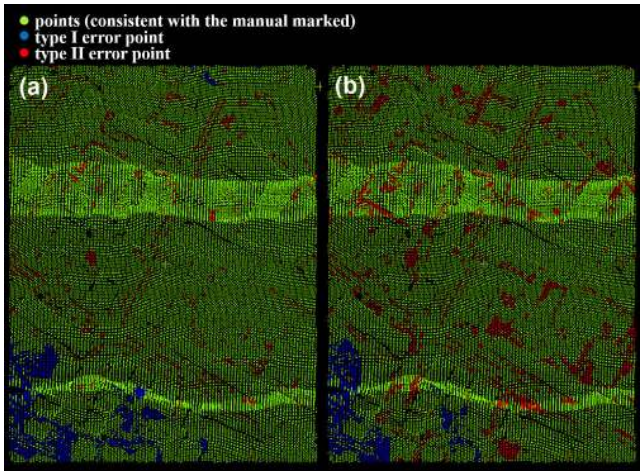

FIGURE 7. Efficiency of classifications. (a) The efficiency of our proposed method. (b) The efficiency of the NURBS fitting method.

According to the position relationship with the generated DTM, the LiDAR points can be separated into the ground points and the non-ground. By marking the same colors with the manual edition, in Fig. 6 we show the accuracy of the proposed method by comparing with the manual classification. In addition, Tab.2 gives the quantitative result of the proposed method: the type I error of our proposed method is 7.10%, the type II error is 10.42% and the total error is 8.65%.

Moreover, the common NURBS fitting method is employed for comparison. For a better result, we rasterize the area of the point clouds, and then select only the lowest point in each raster to ensure that most of the reserved points are ground points. Then the NURBS surface obtained by fitting the processed data is utilized to divide the ground LiDAR points and the non-ground. Fig.7 shows the comparison results in accuracy of the proposed method and the NURBS fitting method. The red and blue points represent the mis-marked non-ground and ground points which are not consistent with the manual marking in Fig.7 and the green points represent the consistent marking. The type I error points (blue points) are mostly concentrated in the lower left corner of the graph, that is, the ground points in that area are mostly misidentified as non-ground points. Low quality of point clouds makes it difficult to classify point clouds in that region.

TABLE 2. Errors of our method and NURBS fitting method (%).

method	Type I error	Type II error	Total error
Our proposed method	7.10	10.42	8.65
NURBS fitting method	4.95	24.69	14.17

The additional rasterizing and selecting process for the NURBS fitting method gains a better performance in type I error comparing with the proposed method. However, the key index of the type II and total error are still much larger. As shown in Tab.2, the accuracy of NURBS fitting method is far worse than the proposed method, though it rewards a bit in the computation complexity.

It can be seen from the experimental results that the proposed method can lead to a high accurate fitting result in the relatively gentle terrain cases. The proposed method can accurately distinguish the roof and tall vegetation, fill in some holes in raw data, and form a complete, smooth surface model. However, for steeper terrain, such as mountainous areas where the slope varies greatly, this method cannot achieve the same performance as the gentle cases. In addition, one of the prerequisites for performing the proposed method successfully and accurately is that ground points should account for most of the experimental data. Thus, for the proposed method, the more exposed surface results in the better effect and the flatter terrain leads to the higher accuracy.

IV. CONCLUSION

This paper proposes a physical process driven DTM construction method based on D-NURBS. In the traditional spline surface fitting process, physical properties are added to make the fitting result more suitable to the surface, thus meet the needs of terrain modeling better. This method is easy to operate and adjust.

The advantages of our method are summarized as follows:

1. The parameters in the experiment have explainable physical meaning and can be adjusted according to the actual situation.
2. The generated spline surface model can not only be used as the basis for filtering the data points, but also be directly used as a DTM. We argue this kind of DTM could eliminate the need to interpolate ground points and fills the gap in the initial data.
3. Since the generation of the spline surface model only needs the information of control points and node vectors, it might show significant advantages in data transmission and presentation.
4. In addition, since this model is based on the dynamic equation of physics, it can also be applied to the dynamic evolution of terrain, such as flood simulation, debris flow monitoring, urban planning and so on.

However, D-NURBS method may not be very effective in processing data with steep slope. When the local shape is extremely complicated, much more control points of the spline surface need to be added to ensure the accuracy of the DTM. As a result, it will make the calculation much more time-consuming and may result in some unstable results. Nevertheless, for such kind of complex terrain, one may adopt partition modeling to reduce the computational complexity while ensuring the accuracy of the model. Consequently, the partitioning, modeling and stitching of terrain data should be the focus of our future work.

REFERENCES

[1] A. Kobler, N. Pfeifer, P. Ogrinc, L. Todorovski, K. Oštir, and S. Džeroski, "Repetitive interpolation: A robust algorithm for DTM generation from Aerial Laser Scanner Data in forested terrain," *Remote Sens. Environ.*, vol. 108, no. 1, pp. 9–23, May 2007.

[2] I. Hurkxkens and M. Bernard, "Computational terrain modeling with distance functions for large scale landscape design," *J. Digit. Landscape Archit.*, pp. 222–230, Apr. 2019.

[3] M. H. Wimmer, N. Pfeifer, and M. Hollaus, "Automatic detection of potential dam locations in digital terrain models," *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 4, p. 197, Apr. 2019.

[4] T. Goodbody, N. Coops, T. Hermosilla, P. Tompalski, and G. Pelletier, "Vegetation phenology driving error variation in digital aerial photogrammetrically derived terrain models," *Remote Sens.*, vol. 10, no. 10, p. 1554, Sep. 2018.

[5] X. Kang, J. Li, and X. Fan, "Line feature extraction from RGB laser point cloud," in *Proc. IEEE 11th Int. Congr. Image Signal Process., BioMed. Eng. Inform. (CISP-BMEI)*, Oct. 2018, pp. 1–5.

[6] O. S. Azeez, B. Pradhan, and R. Jena, "Urban tree classification using discrete-return LiDAR and an object-level local binary pattern algorithm," *Geocarto Int.*, pp. 1–19, Oct. 2019.

[7] X. Lai, J. Yang, Y. Li, and M. Wang, "A building extraction approach based on the fusion of LiDAR point cloud and elevation map texture features," *Remote Sens.*, vol. 11, no. 14, p. 1636, Jul. 2019.

[8] Z. Chen, B. Gao, and B. Devereux, "State-of-the-art: DTM generation using airborne LIDAR data," *Sensors*, vol. 17, no. 12, p. 150, Jan. 2017.

[9] G. Vosselman, "Slope based filtering of laser altimetry data," *Int. Arch. Photogramm. Remote Sens.*, vol. 33, pp. 935–942, Jul. 2000.

[10] J. Silván-Cardenás and L. Wang, "A multi-resolution approach for filtering LiDAR altimetry data," *ISPRS J. Photogramm. Remote Sens.*, vol. 61, no. 1, pp. 11–22, Oct. 2006.

[11] G. Sithole and G. Vosselman, "Automatic structure detection in a point-cloud of an urban landscape," in *Proc. IEEE 2nd GRSS/ISPRS Joint Workshop Remote Sens. Data Fusion Urban Areas*, May 2003, pp. 67–71.

[12] F. Yuan, J. X. Zhang, L. Zhang, and J. X. Gao, "DEM generation from airborne LIDAR data," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 38, nos. 7/C4, pp. 308–312, 2009.

[13] X. Meng, L. Wang, J. L. Silván-Cárdenas, and N. Currit, "A multi-directional ground filtering algorithm for airborne LIDAR," *ISPRS J. Photogramm. Remote Sens.*, vol. 64, no. 1, pp. 117–124, Jan. 2009.

[14] M. R. James and S. Robson, "Straightforward reconstruction of 3D surfaces and topography with a camera: Accuracy and geoscience application," *J. Geophys. Res., Earth Surf.*, vol. 117, no. F3, 2012.

[15] J. Kilian, N. Haala, and M. Englich, "Capture and evaluation of airborne laser scanner data," *Int. Arch. Photogramm. Remote Sens.*, vol. 31, pp. 383–388, Jul. 1996.

[16] X. Meng, L. Wang, and N. Currit, "Morphology-based building detection from airborne Lidar data," *Photogramm. Remote Sens.*, vol. 75, no. 4, pp. 437–442, Apr. 2009.

[17] P. Lohmann, A. Koch, and M. Schaeffer, "Approaches to the filtering of laser scanner data," *Int. Arch. Photogramm. Remote Sens.*, vol. 33, pp. 540–547, Jul. 2000.

[18] K. Zhang, S.-C. Chen, D. Whitman, M.-L. Shyu, J. Yan, and C. Zhang, "A progressive morphological filter for removing nonground measurements from airborne LIDAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 4, pp. 872–882, Apr. 2003.

[19] R. A. Haugerud and D. J. Harding, "Some algorithms for virtual deforestation (VDF) of LIDAR topographic survey data," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 34, no. 3/W4, pp. 211–218, 2001.

[20] P. Axelsson, "Processing of laser scanner data—Algorithms and applications," *ISPRS J. Photogramm. Remote Sens.*, vol. 54, nos. 2–3, pp. 138–147, 1999.

[21] D. Mongus and B. Žalik, "Parameter-free ground filtering of LiDAR data for automatic DTM generation," *ISPRS J. Photogramm. Remote Sens.*, vol. 67, pp. 1–12, Jan. 2012.

[22] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D point cloud based object maps for household environments," *Robot. Auton. Syst.*, vol. 56, no. 11, pp. 927–941, 2008.

[23] D. Terzopoulos and H. Qin, "Dynamic NURBS with geometric constraints for interactive sculpting," *ACM Trans. Graph.*, vol. 13, no. 2, pp. 103–136, Apr. 1994.

DANLEI YE is currently pursuing the Ph.D. degree with Beihang University. Her research interests include CAD and CG.

XIN JIANG is currently an Associate Professor with Beihang University. His research interests include data processing and CAM.

GUANYING HUO is currently an Assistant Professor with Beihang University. His research interests include design and manufacturing of freeform surface.

CHENG SU is currently pursuing the Ph.D. degree with Beihang University. His research interests include CAD and CAM.

ZEHONG LU is currently pursuing the Ph.D. degree with Peking University. His research interests include computational geometry for CAD/CAM.

BOLUN WANG is currently pursuing the Ph.D. degree with Beihang University. His research interests include CAD and CG.

ZHIMING ZHENG is currently a Professor with Beihang University. His research interests include ODE and dynamical systems.