



A novel approach for surface defect detection of lithium battery based on improved K-nearest neighbor and Euclidean clustering segmentation

Xinhua Liu¹ · Lequn Wu¹ · Xiaoqiang Guo¹ · Dariusz Andriukaitis² · Grzegorz Królczyk³ · Zhixiong Li³

Received: 10 January 2023 / Accepted: 28 April 2023 / Published online: 19 May 2023
© The Author(s) 2023

Abstract

Surface defects of lithium batteries seriously affect the product quality and may lead to safety risks. In order to accurately identify the surface defects of lithium battery, a novel defect detection approach is proposed based on improved K-nearest neighbor (KNN) and Euclidean clustering segmentation. Firstly, an improved voxel density strategy for KNN is proposed to speed up the effect for point filtering. Then, the improved clustering segmentation strategy is applied to distinguish point clouds with defect features. The outline fitting algorithm based on the least square method is applied to determine geometric features of each surface defect which are used to classify defect types. Furthermore, experimental results show that the proposed surface defect detection method reaches 99.2% accuracy and 35.3-ms average time consumption for data processing. Finally, an industrial application example of lithium battery production is demonstrated, which meets the requirements of industrial application. All these reports exhibited that the industrial visual inspection system with rapid measurement is an effective method and guarantees for accelerating industrial production and manufacturing in the future.

Keywords Lithium battery · Intelligent defect detection · Machine learning · Computer vision

1 Introduction

Nowadays, with the increasing demand for electricity, lithium battery has become the most commonly used electric energy storage device due to its advantages of lightweight and high energy density [1–3]. However, many new energy vehicle and electric tools with lithium battery are usually damaged because of the integrity of the battery system in the process of complex industrial production [4]. Moreover, many safe accidents in daily scenario are caused by defective lithium batteries that are due to the limitations

of the detection method. Thus, an efficient identification approach for surface defect detection is a challenging issue and decides whether the lithium battery can be manufactured with high efficiency and low risk.

Due to the poor working conditions of defect identification, direct identification approaches for surface defect detection are hard to be checked permanently and efficiently, so many indirect identification methods are proposed through the image information of industrial machine vision reflecting surface defect detection such as 2D industrial machine vision and 3D industrial machine vision [5–7]. However, the disadvantage of 2D industrial machine vision is being unable to obtain the height information. The 3D surface reconstruction of the object can be detected by the extraction of 3D point cloud features. Therefore, it becomes an active research field by using machine vision technology to identify surface defects in industries.

Many researchers have worked on the problem of surface defection and proposed different solutions [8–10]. However, most of them have worked on the qualitative surface defect detection and few researchers study the quantitative detection of surface defect detection. To the best of

✉ Zhixiong Li
z.li@po.edu.pl

¹ School of Mechatronic Engineering, China University of Mining and Technology, Xuzhou 211006, China

² Department of Electronics Engineering, Kaunas University of Technology, Studentu St. 50-438, 51368 Kaunas, Lithuania

³ Faculty of Mechanical Engineering, Opole University of Technology, 45-758 Opole, Poland

our knowledge, the quantitative detection of surface defect detection has almost not been dealt with. Considering the roughness and height of surface defect detection, a 3-dimensional visual measurement system would be an effective way to solve the problem. Based on the past work on K-neighbors algorithm [11] and Euclidean clustering segmentation [12], this paper tries to tackle this problem.

Bearing the above observation in mind, we apply an integration approach of improved K-neighbors algorithm and Euclidean clustering segmentation to solving the problem of surface defect detection and the rest of this paper is organized as follows. Surface defect detection approaches and 3D point data processing methods are reviewed in section 2. Section 3 proposes the integration approach of improved K-neighbors algorithm with voxel density strategy, and introduces Euclidean clustering segmentation optimized with separation degree parameter to distinguish height feature region and improve segmentation algorithm. Section 4 provides an experimental example by selecting the optimal algorithm parameters and carries out the surface defect recognition to verify the accuracy, efficiency, and outperforming. An industrial example of lithium battery automation production based on the proposed system was demonstrated to specify the application effect in Section 5. Our conclusions and future work are summarized in Section 6.

2 Literature review

Recent publications relevant to this paper are mainly concerned with two research streams: surface defect detection approach and 3D point data processing method. In this section, we try to summarize the relevant literatures.

2.1 Surface defect detection approach

For surface defect detection approach, many researchers had worked on the problem and proposed different solutions for the last decades. The traditional two-dimensional vision algorithm uses a series of algorithms such as threshold [13], filter [14], and morphology [15] to analyze the image, and uses the geometric shape and gray difference of the defects to recognize and classify the defects. The following scholars have contributed to this field. Liu Y et al. [16] proposed an automatic flaw inspection scheme with rapid detection of the defects in the potential defect region extracted from the ROI image for online real-time detection on LIBE surface. Liu Z. Y. et al. [17] proposed a deep-learning method based on the deformable convolution and concatenate feature pyramid neural networks for improving the inspection to adaptively extract the features of defect shape and location. Hao et al. [18] proposed a steel surface defect inspection network to extract complex features from multi-shape steel surface

defects using deformable convolution. In Zhao et al. [19], an improved 3D laser image scanning system (3D-LDS) was established to improve the inspection accuracy using a deep CNN architecture combined fully connected networks and fully convolutional network. The 2D image does not provide enough information to enable recognition between different typologies of defects, and from the 3D point cloud features are extracted to detect and classify the defect. Yan et al. [20] constructed a composite vision system enabling simultaneous 3D-depth information and 2D-gray imaging features to obtain the welding bead boundary reliably and accurately. Carlos A. Madrigal et al. [21] propose a novel 3D local descriptor called the Model Point Feature Histogram (MPFH) for defect detection. Qian et al. [22] propose a real-time 360° 3D surface defect inspection approach based on fringe projection profilometry without any auxiliary equipment for position control.

2.2 Point cloud data processing method

Traditionally, machine learning algorithms [23–25], such as K-nearest neighbors (KNN), support vector machine (SVM), and K-means clustering methods, and some regression algorithms were used to solve the problems with classification and clustering. In fact, as a perfect integration approach of K-neighbors filtering algorithm and Euclidean clustering segmentation, 3D data processing could solve complex identification systems in a manner of high-defect processing efficiency, fine local characteristics, and better generalization performance compared with traditional machine vision detection. As a vital task of data calculation, the classification algorithm parameters have a great influence on the generalization performance and processing effect, and the process of parameters selection was complex and would consume a vast calculation time. Applying traditional Euclidean clustering methods in point cloud segmentation could cause sub-segmentation and slow the convergence speed [26]. Thus, several AI-based clustering algorithms were proposed to solve this problem such as multi-view-based clustering algorithm, density-based spatial clustering algorithm, depth-neural-network-based clustering algorithm, etc. Dhanachandra et al. [27] proposed clustering technique is based on the density estimation of the surrounding pixel values. Lei et al. [28] proposed an image segmentation algorithm based on improved fuzzy clustering. Zhou et al. [29] proposed an unsupervised segmentation framework based on a novel deep image clustering (DIC) model that consists of a feature transformation subnetwork (FTS) and a trainable deep clustering subnetwork (DCS) for unsupervised image clustering. Liu H. et al. [30] put forward a multi-objective fuzzy clustering algorithm with multiple spatial information (MFCMSI) for noisy color image segmentation.

2.3 Discussion

However, although many approaches to detect and recognize surface defects and 3D reconstruction of piece methods for surface defect problems have been proposed in above literatures, they have some common disadvantages summarized as follows. Firstly, the usual method through 2D image to reflect the change of surface defects is not possible to perform precise metrology of the defect and it is needed to determine the appropriate lighting system in the specific domain of an application for a new detection of identification. Secondly, the methods through 3D reconstruction of piece and 3D point data processing to identify the surface defects are only realized on the character remodeling, and these data information usually have not been extracted completely from the 3D surface model. Finally, few researches have focused on the three-dimensional defect quantification for surface defect detection.

In this paper, an integration approach based on improved K-neighbors algorithm and Euclidean clustering segmentation is proposed to identify the surface defects. A simulation example and a comparison with other algorithms are carried out, and the proposed approach is proved feasible and efficient.

3 The proposed approach

The solution of defect detection system is illustrated in Fig. 1 to recognize surface defects. Our system began with obtaining the depth image by the structured light system; and as a result, the 3D point cloud model is obtained by the depth image (Fig. 1a), followed by the calculation of the model that filter the point cloud data (Fig. 1b), and then segment the model by European clustering algorithm (Fig. 1c), that fit each region using the least squares to estimate and quantify the defect information (Fig. 1d). By

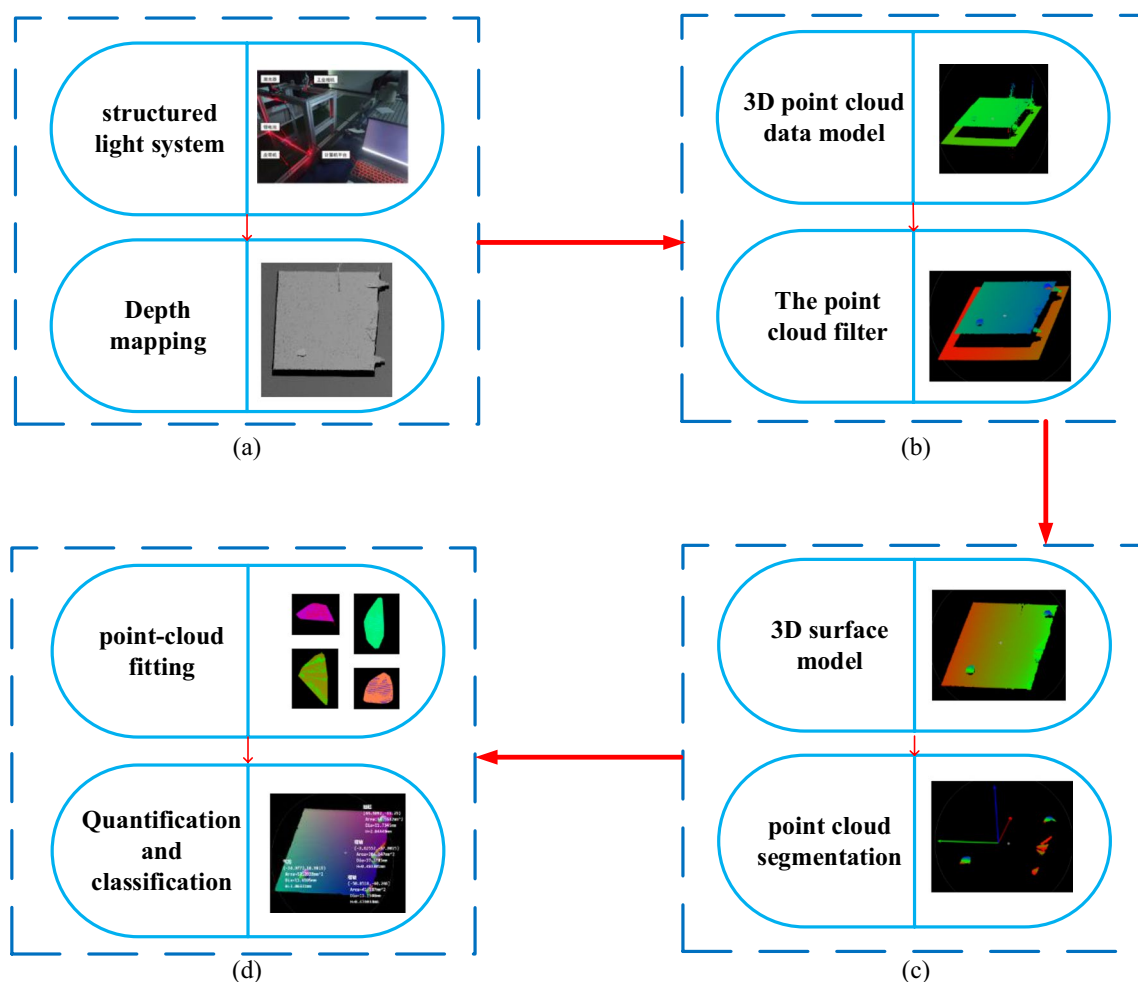


Fig. 1 The framework of the proposed surface defect detection approach: **a** image point cloud data acquisition; **b** point cloud filtering; **c** point cloud segmentation; **d** point cloud quantization classification

quantifying defect information of the coordinate center position, defect area, defect height, and shape size, each point on the 3D surface is classified into one of four types of defects-bubble, fold, warping, and pit. In this work, it is considered that with this set of primitives it is possible to describe any typology of the defect. Finally, from these regions are extracted 2D features to recognize the defect in a classification stage.

The flowchart of the proposed approach for improved KNN filtering algorithm and improved Euclidean clustering algorithm in defect identification is shown in Fig. 2.

3.1 Image acquisition

The depth image is obtained from the 3D structured light system. We use a (CMOS) camera (Teledyne DalsaG3-GM12-M2590, On-Semi Python 5000 p1) to capture the laser fringes moving on the mobile platform, which are emitted by a line laser of our choice, and the height information is stored in the sheet of light model of the computer terminal, and then the depth image of the target image is obtained. This depth image acquisition and the 3D reconstruction system are shown in Fig. 3. The specific parameters of the hardware are shown in Table 1.

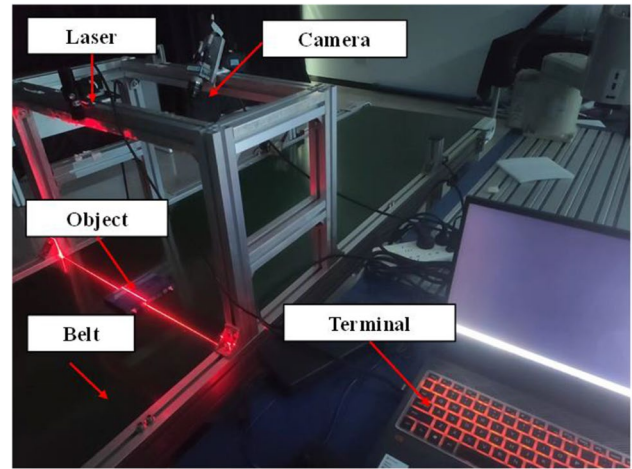


Fig. 3 Scheme of the 3D reconstruction system

3.2 Improved K-neighbors algorithm

There are two problems in the process of three-dimensional data model acquisition. One of the problems is that there is a lot of point cloud data in the model, which affects the speed of data processing, and another problem exists with some inevitably noisy data, e.g., outliers, which need to be

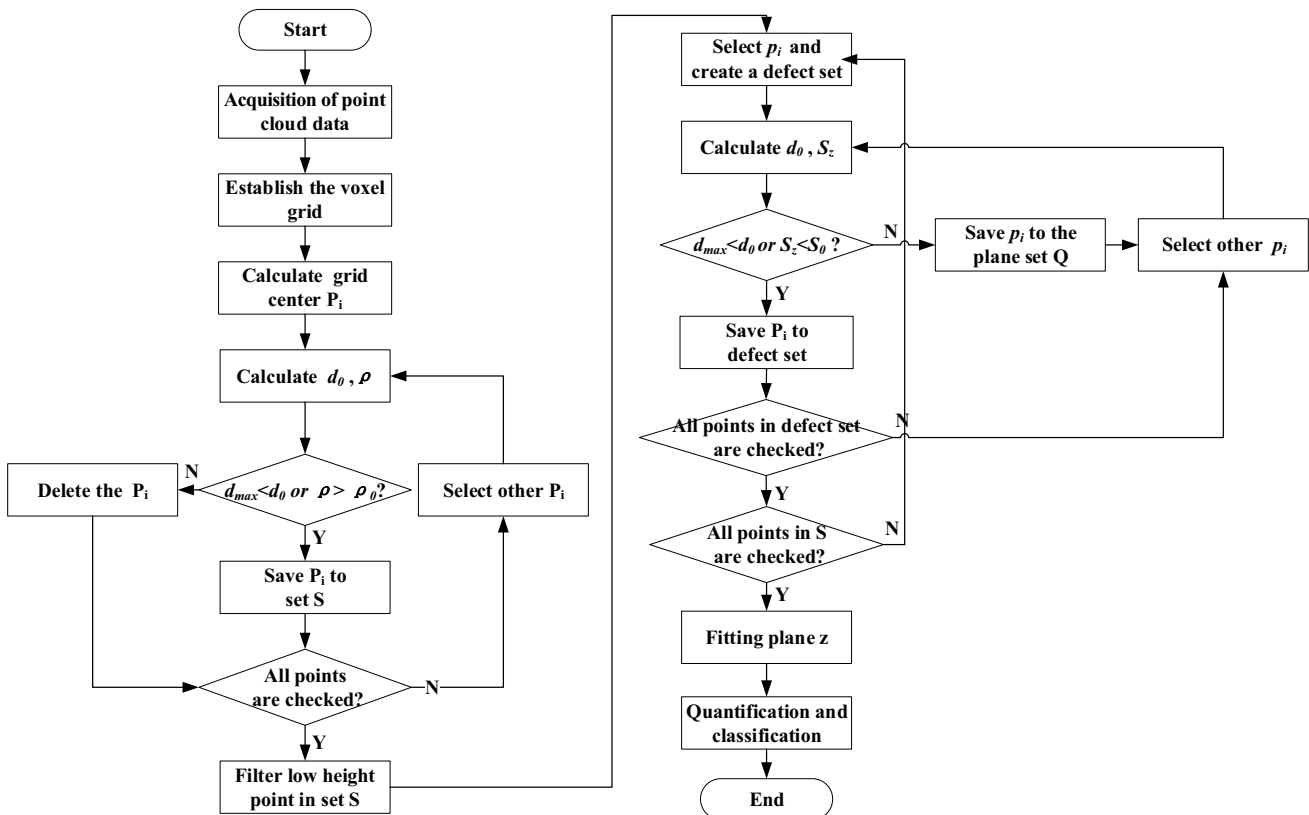


Fig. 2 The flowchart of the proposed approach

Table 1 Main equipment specifications

Device	Specifications
Camera	Teledyne DalsaG3-GM12-M2590
Sensor size	4.8μm×4.8μm
Image resolution	2592×2048
Lenses	$f = 25\text{mm}$
Laser	150mw
Software	C#, halcon19, and VS2019

removed to ensure the integrity of the capture data model. Therefore, if we use voxels as computational units (Fig. 4a) to handle voxel grids, the processing speed of K-neighborhood and the accuracy of outlier processing will be greatly improved. Outlier as a form of noise can easily be an inlier of structures and generate a very large number of structures.

Given a point cloud of the sample 3D surface called S set, the S sets are represented by a voxel grid cube with side length l_0 that can greatly reduce the efficiency of point cloud data processing. P_i represents the grid coordinate which is the center of gravity of point cloud data in a unit; let $P_i \in S$ be point on the surface. The K nearest neighbors include neighbors P_{ij} in the sphere with radius d_0 which are represented by set Q_k . The problem of choosing d_0 affects the size of filtering area and the number of K-adjacent points. The problem of choosing the number k and distance d_0 is called the correct scaling factor, which affects the estimation of outliers. The correct outlier estimation can effectively remove the model noise. The selection of parameter k determines whether the point is an outlier in Q_k domain directly. These outliers are far away from the model and can be effectively filtered by setting d_0 and k parameters (Fig. 4b).

However, there are some sparse noises generated by reflection in data acquisition, which are close to the main

body of the model and cannot be removed effectively. In order to correctly judge whether the set of point clouds represented by the voxel grid is a sparse point group, we introduce the density parameter ρ (Fig. 4c), which represents the voxel density of the voxel grid in the region Q_k . This value reflects the number of point clouds contained in the voxel, indirectly identifying and filtering noise. In Fig. 4 it is shown how to estimate P_i with ρ .

The structure restricted to the l_0 parameter is defined as a set of voxels, where point P_i is a collection of solid points p_{ij} , and voxel grids represent surface variations in a simple manner. Region $Q_k = \{p_i\}$ ($i = 1, 2, \dots, k$) is represented by a k structure, where the parameters p_{ij} and n are used to represent this P_i structure. Therefore, we propose a method to evaluate the voxel structure P_i in the Q_k region, where the concept of voxel density is added to the measure of P_i , and then their structure density ρ is calculated. This is shown in Eq. (1), where n_i represents the number of point clouds in the structure P_i found on the surface. Figure 4 shows the density estimation method for the voxel point P_i . By setting a density threshold, we estimate the density of the noise near the real model. The density value ρ directly reflects the proportion of the voxel grid in the set Q_k ; this indirectly illustrates the sparsity of the points within the voxel grid, which should be filtered out. In Fig. 4c, the density of P_i will be defined.

$$\rho = \frac{n_i}{(n_{ij})_{max}} \tag{1}$$

The effect of voxel density can be seen in Fig. 4c. If the number of points in the P_i Cloud n_i is 2 (blue grid) and the maximum $(n_{ij})_{max}$ in the Q_k neighborhood is 6 (purple grid), then the density ρ_i of P_i points in the Q_k set is 1/3. If the threshold ρ_0 we set is greater than ρ_i , then the P_i point will be considered sparse and eliminated.

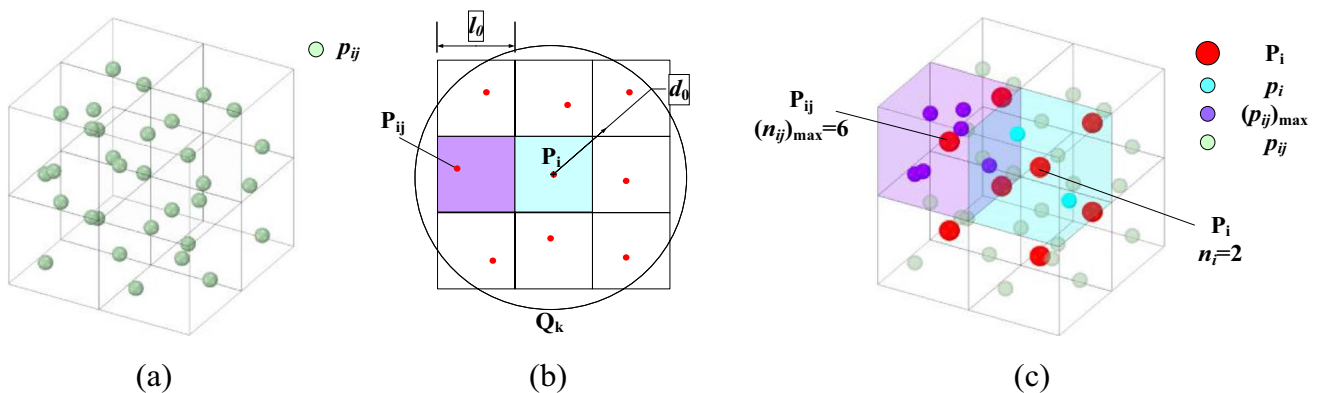


Fig. 4 Improved K-neighbors algorithm: **a** the voxel grid; **b** the parameter of KNN algorithm; **c** density parameter

3.3 Clustering segmentation

As can be seen in Fig. 5, the defect segmentation is divided into three parts: data acquisition, data processing, and results display. Considering that the defects exist separately, we need to use Euclidean clustering algorithm to segment the model, but the defects near the location cannot be distinguished because of sub-segmentation. For this situation, we study the existence of plane region in under-segmentation. Segmentation with an undistinguishable under-segmentation region, as the most important processing link, needs to be further improved. Comparing the feature of defects with a flat area and a defect area, the degree of point-z separation S_z would be introduced to further segment the defects.

Firstly, the seed point p_i of the spatial points in region R is selected. In this paper, we take any point p_i in the under-partitioned region and obtain all points $p_i \in S, S = \{ p_i \in 1, 2, \dots, n \}$ by spatial index within a certain distance. We get the z value for each p_i point, which is its height value. Besides, in the process of parameter evaluation, we introduce the degree of point-z separation S_z as a feature to evaluate the seed point p_i , where the formula of S_z is shown in Eq. (2). We set S_0 , and if the value of S_z is less than S_0 it is considered to be a plane set, and so on. All point clusters could be divided into a plane set (Q) and a defect set (A, B, ...) with the feature of height difference, so that the segmentation is finished. At last, different colors are used to represent defects segmentation results.

$$s_z = \sqrt{\frac{\sum_1^n (z_i - z_{ave})^2}{n - 1}} \tag{2}$$

where z_{ave} denotes average height, $z_{ave} = \frac{\sum_1^n z_i}{n}$. The standard deviation S_z is introduced to describe the discreteness of point p_i in region R. When the query point p_i is located on a flat plane, the height difference in region R is small, and the smoothness S_z is small too. That means region R is the smoother, and the standard deviation S_z is smaller.

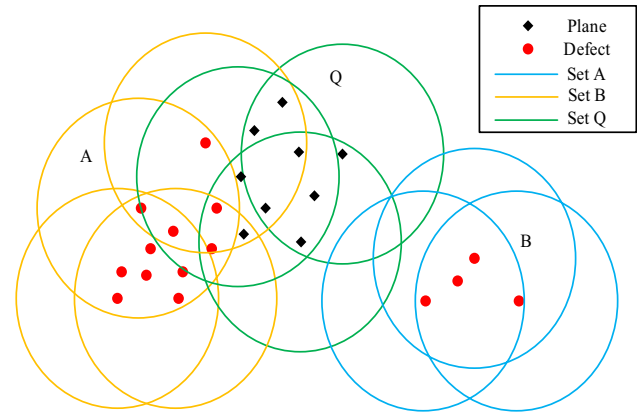


Fig. 6 The relationship of point cloud in clustering segmentation

Therefore, by judging the value of S_z , we separate the point cloud set with the plane features and detect the defects with complex height variations from the improved Euclidean segmentation to solve the problem of under-segmentation.

Figure 6 expresses the clustering relationship between defect sets and plane sets. There may be plane set point cloud data in defect A set, which needs to be further divided into plane set Q by using the threshold S_z . Set Q as a separate category includes smoothed points that would not be considered as a defect set A, and eventually filtered set Q out while retaining the remaining clustering results A and B set.

3.4 Quantification and classification

The final task is to fit the point cloud data and quantify the defect information, after segmentation is completed. Four kinds of defects with height feature would be classified by the necessary information of 2D features such as geometric shape defect size defect and height. As a solution, we choose “plane” primitives to fit the defect. We assess and categorize defects with high-profile features, focusing on bubbles, folds, warps, and pits. In space

Fig. 5 The proposed point clouds segmentation algorithm for surface defect detection

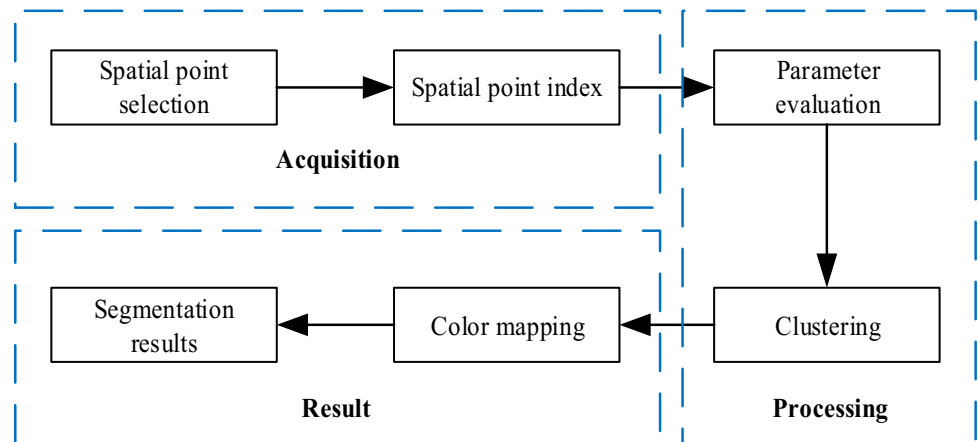


Table 2 Classification criteria

Types	Internal defects	The external defects	Height	Diameter
Fold	√	-	$h > 0$	$d > 12$
Bubble	√	-	$h > 0$	$0 < d < 12$
Pit	√	-	$h < 0$	-
Fold	-	√	$0 < h < 2$	-
Warping	-	√	$h > 2$	-

coordinates, the expression for a plane can be written as Eq. (3).

$$z = a_0x + a_1y + a_2 \tag{3}$$

Least squares are used to fit a plane with discrete points in defect set, that is, to find a plane $z = a_0x + a_1y + a_2$ that is closest to each point. According to the least squares, the deviation $Q = \sum (a_0x + a_1y + a_2)^2$ is the smallest. That is to say, we require a set of a_0, a_1, a_2 to fit plane so that Q is the smallest value for the given discrete points. Setting the first derivative of S to 0, we could get Eq. (4).

$$\begin{bmatrix} \sum x_i^2 & \sum y_i x_i & \sum x_i \\ \sum x_i y_i & \sum x_i^2 & \sum y_i \\ \sum x_i & \sum y_i & n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum z_i x_i \\ \sum z_i y_i \\ \sum z_i \end{bmatrix} \tag{4}$$

Then we project the point cloud into the fitting plane, and use the edge projection technique to get the plane projection of the defect fitting, so as to calculate the defect area, the maximum diameter of the defect, the center of gravity of the defect, and so on. It should be noted that the fitted defect plane cannot accurately represent the actual shape and height of the defect, but it can also be used to obtain other useful information about the defect. The useful characteristics such as area and maximum diameter of the defect, and the center of gravity coordinates will vary depending on what the fitting plane looks like. For the height information, the maximum height difference

between the z-axis coordinates of the point cloud is determined to indicate the height of the defect.

In the process of defect classification, four types of defects with the coordinate information are divided into external and internal defects. Along the edge of an object’s surface position is used as the boundary between external and internal parts. The external part has fold and warping defect that could be distinguished by the difference of their own height, the lower for the fold, the reverse is warping. The internal part can be divided into fold, bubble, and pit, in which we can judge whether it is a pit according to the height value of positive and negative, and judge whether it is a bubble according to the size characteristics of the defects. After analysis of lithium-ion batteries, Table 2 classification criteria will be set. According to the coordinates of the center point of the defect, the defect boundary value is 10 mm which is used to judge whether it is on the edge of the tested object. In the former case, if the defect is outside, the defect with height h more than 2 mm is a warping, and the other situation is a fold. In the latter case, if the defect is inside, the defect with height h less than 0 is a pit. Otherwise, if the defect with height h is more than 0, the defect with diameter d less than 12 mm is a bubble and its diameter d more than 12 is a fold.

4 Experimental example and discussion

In this section, an experimental example was put forward to verify the feasibility and efficiency of the proposed approach. The relation model of clustering segmentation, classification, and defect quantification was established and the surface height defect could be effectively identified by the quantitative defect identification in through the relation model. Then, the result of the experiment was provided to estimate the efficient and accurate of surface defect recognition proposed in this paper.

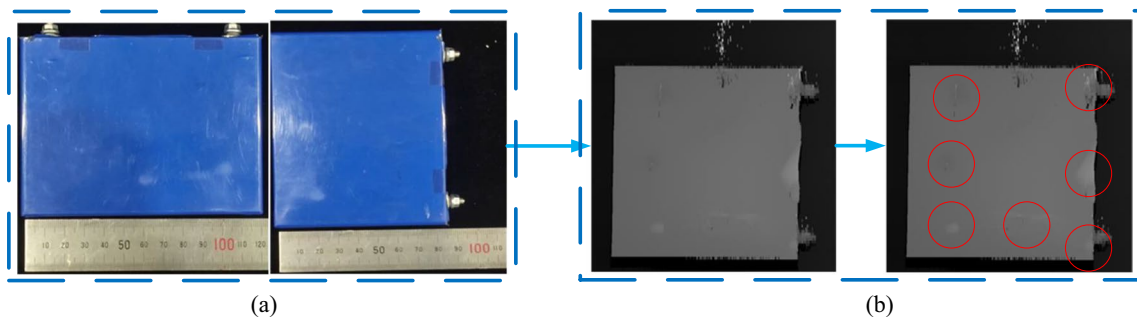


Fig. 7 Lithium battery and its depth image: **a** experimental lithium battery; **b** depth image and defects

Table 3 The sampling point of the partial data

Index	x-coordinate	y-coordinate	z-coordinate
5355	-118.66	165.816	110.475
5356	-118.745	165.816	110.470
5357	-118.83	165.816	110.475
5358	-118.915	165.816	110.472
5359	-119.0	165.816	110.412
5360	-119.085	165.816	110.464
5361	-119.17	165.816	110.466
5362	-119.255	165.816	110.475
5363	-119.34	165.816	110.475
5364	-119.425	165.816	110.475

4.1 Sample data preparation

Laboratory bench is used to acquire data by the 3D structured light scheme, whose goal is to get a depth mapping. In this experiment, we chose a lithium battery with size of $120 \times 85 \times 12$ mm. In order to intuitively understand the actual state of the object to be measured, the lithium battery is shown in Fig. 7a. Next, the abovementioned structured light system is used to get the 3D image. Figure 7b indicates using the structured light system to get the depth image. And it already contains the defect feature needed to further segment the defect. These defects on the depth diagram are shown in red circles.

Few defects can be seen and that cannot be recognized effectively. Furthermore, we obtain the 3D model by stretching the z-direction of the depth image. Parts of the point cloud coordinates are shown in Table 3. These coordinate points need to be further process by algorithm for the desired target defect data.

4.2 Parameters for the simulation example

There are five parameters for the improved KNN algorithm and the segmentation algorithm. In this section, we discuss the parameters of the two algorithms and choose the most suitable parameters for this experiment.

For the improved KNN filtering algorithm, three parameters of d k ρ need to be selected. In the simulation experiment, the number of the total point cloud data of our model is 133,666, and the relationship between the total points and the influence of the three parameters is shown in Fig. 8. For the neighbors number k , it is obvious that the value represents the number of point clouds that reflects the density of the point in region Q. If distance d_0 of Q is greater than 20 (Fig. 8a), the number of total points decreases precipitously that reduces the processing effect. For the selection of parameter d , we have to consider the density of point cloud data collection. When the distance is too small, the dense point could be selected, but some defects points will lose; when the distance is too large, it is difficult to select all outliers that the values of distance d need to achieve a balance (Fig. 8b). And finally, the density ρ of voxel grid is to be determined removing sparse voxels from edges, which reflects the density of the point cloud in the k -neighborhood of the voxel grid (Fig. 8c). Therefore, in this simulation example, the parameters of the proposed algorithm were configured as follows: $N=12$, $d=1$, $\rho=0.5$.

In order to select the segmentation parameters, we obtain the data of the under-segmentation region, and the z-axis coordinate change corresponding to the sequence point cloud data is shown in Fig. 9a. The point cloud whose height is less than 0.7 in the index range from 0 to 4400 is considered to be a flat point that needs to be removed by the improved algorithm, but there are defect points whose Z-axis height is more than 0.7 in the index range from 4400 to 5402, which should remain as defects.

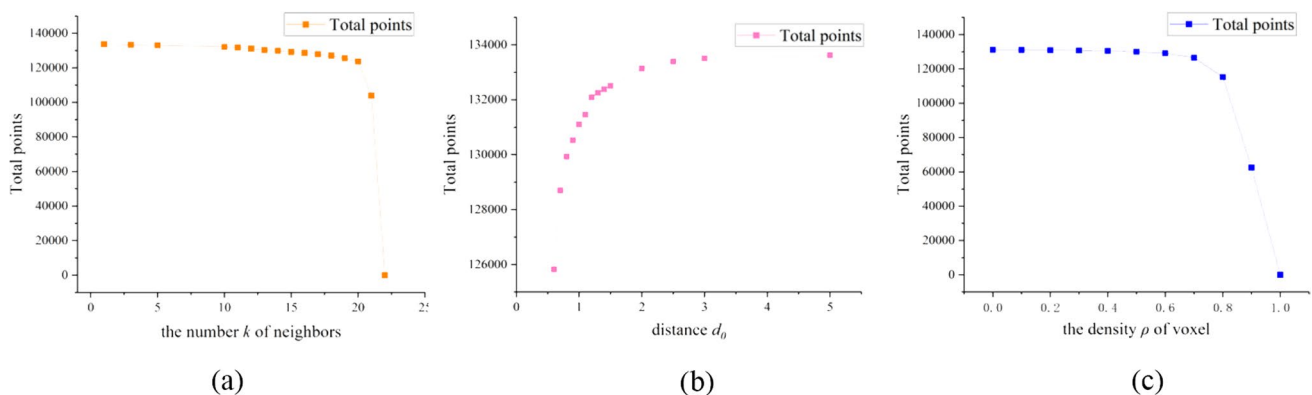


Fig. 8 The influence of three parameters on the filtering effect: **a** the influence of parameter k on the filtering effect; **b** the influence of parameter d_0 on the filtering effect; **c** the influence of parameter ρ on the filtering effect

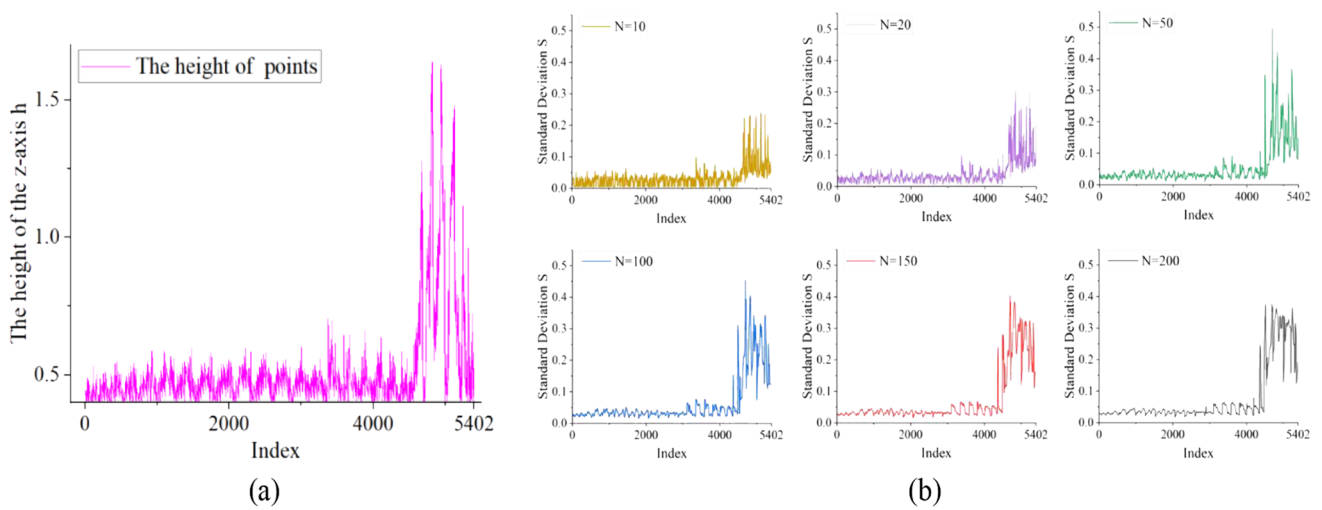


Fig. 9 The height of points and the influence of parameter N selection on standard deviation S : **a** the height of points in the sub-segmentation region; **b** the influence of parameters N selection on standard deviation S

As shown in Fig. 9b, we need to select the partition parameters n and S_z . The parameter n represents the neighborhood range of the selected point, which determines the assessment range of the point. Considering that the data collected by the image are separated in strips where there are gaps in the middle, strips will affect the results of the parameter range evaluation. For this reason, the value of n should be chosen as large as possible so that the change of points in the Z-axis near can be better expressed. On the contrary, if the n value is too large, it will increase the computational complexity and reduce the efficiency of processing, which need to achieve a balance. Only the point of uneven region can be clearly expressed by standard deviation S during the selection of n value.

The standard deviation S_z reflects the discreteness of n points along the Z-axis in the region R, and the modified value indirectly reflects the roughness of the points. The relationship between sequence point data and S_z under

different conditions is shown in Fig. 9b. After observation, it can be found that when $N = 150$, rough areas ($S_z > 0.1$) and smooth areas could be distinguished by $S_z = 0.1$. As a result, in this experiment, the parameters of the proposed algorithm were configured as follows: $N = 150$, $S_z = 0.1$.

4.3 Simulation results

During the simulation, we divide voxel grids from the point cloud data and delete the sparse and outliers using an improved KNN algorithm. The processing of improved algorithm is shown in Fig. 10. The blue regions represent outliers that are far away from the model body. In the red circle are the sparse point clouds filtered by the improved algorithm. These point clouds exist at the edge of the model and are difficult to be recognized by the traditional KNN algorithm. The results show that the algorithm can filter outliers and sparse points correctly.

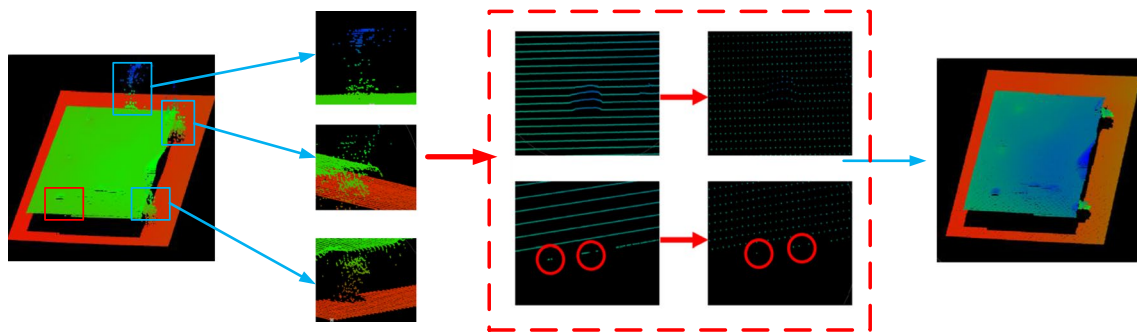


Fig. 10 The effect of improved KNN algorithm

Table 4 Compare with other algorithms

	KNN	Voxel-based KNN	The proposed method
Number of points	131,826	131,101	129,980
Time consumption (ms)	23	6	9

In order to prove the superiority of the proposed method, we compared the efficiency and effectiveness for several types of KNN algorithms in Table 4. The software’s commands are used to calculate the average time consumption of running the proposed algorithm from the beginning of the algorithm to the end of the algorithm by multiple times. The result shows that voxel-based KNN algorithm is faster which takes about 6 ms. Though the density parameter is added at the expense of some speed, for the sparse voxel-based points, the processing effect is better and the processing speed is more ideal. Therefore, the improved KNN algorithm we proposed is excellent.

To solve the problem of under-segmentation, we use the parameter standard deviation S of z-axis to improve the

Euclidean segmentation algorithm. The feature of defects with highly variable will be reclassified and delete the data of flat point cloud with standard deviation. The purple arrow shows this process in Fig. 11, which resolves the region under-segmentation problem in standard deviation to obtain highly defective regions. And the result of segmentation is displayed in different colors.

The quantization results of the feature information extracted from the least squares fitting of the defect plane and the point cloud of the defect itself are shown in Fig. 12. The feature information included defect coordinates, defect area, defect height difference, and defect maximum size. According to the classification criterion in Table 2, we classify the defects by the information obtained from the fitting defects. Next, the feature information will be the most important basis for determining the type of defect. Table 5 shows the results of defect quantification and classification.

The result of defect information quantification with type of defects and defect primitives can be seen clearly in Table 5. And the precision of defect detection can reach 0.01 mm, which meets the requirements of real-time detection.

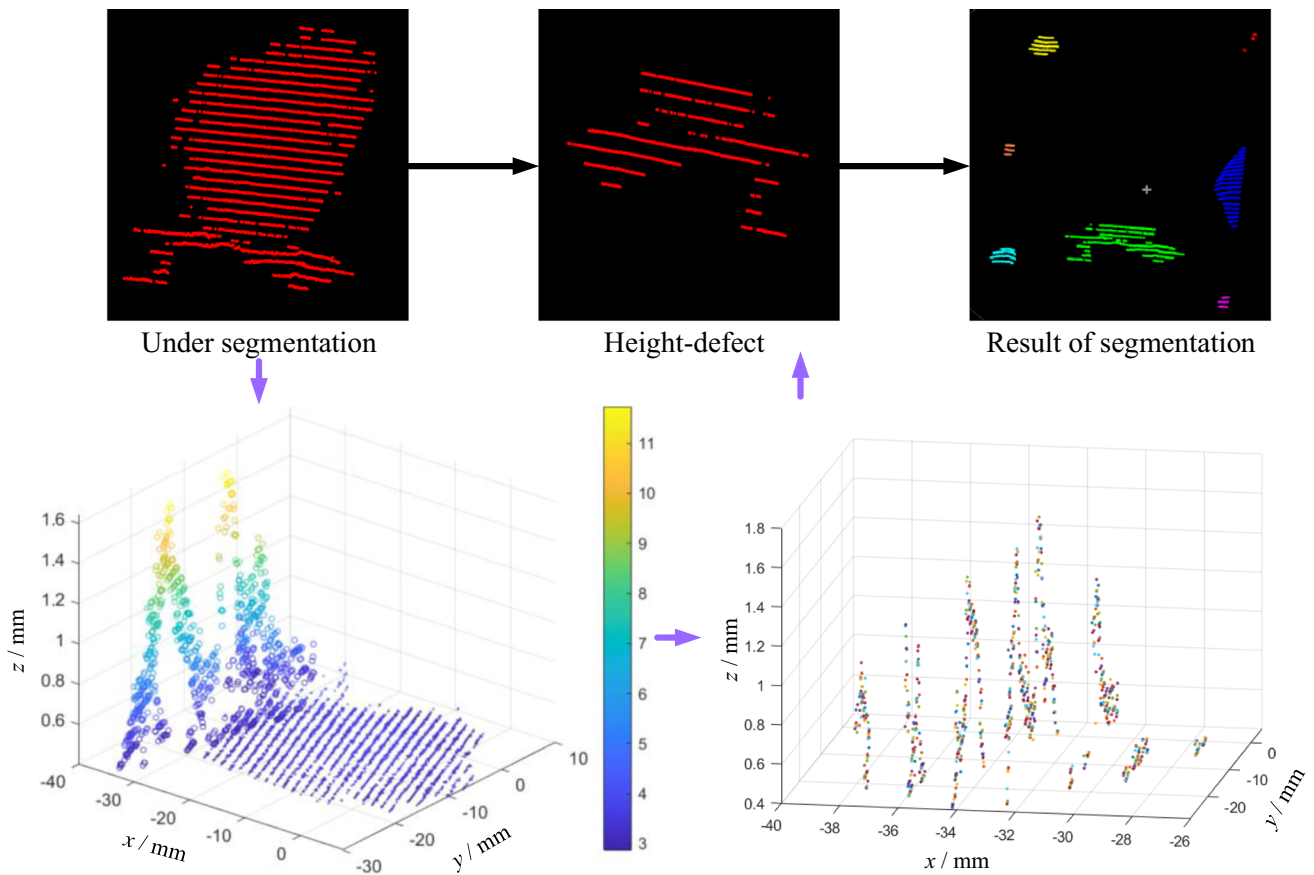


Fig. 11 The result of defect segmentation

Fig. 12 Fitting effect and quantization result. **a** Fitting effect of defect points; **b** quantization result of defect points

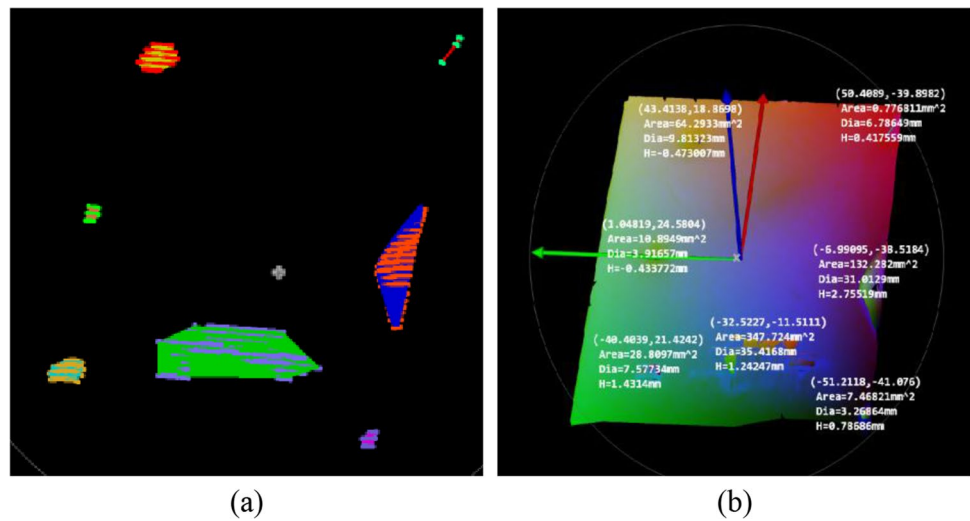









Table 5. Defect quantification and classification results

Index	Coordinate /mm	Area /mm ²	Diameter /mm	Height /mm	Type of defects	Primitives
0	50.409 -39.898	0.777	6.786	0.418	'fold'	
1	-32.523 -11.511	345.219	35.417	1.242	'fold'	
2	-6.991 -38.518	133.691	31.013	2.755	'warping'	
3	-40.404 21.424	34.850	7.577	1.431	'bubble'	
4	-51.212 -41.076	7.468	3.269	0.787	'fold'	
5	43.414 18.870	63.004	9.813	-0.473	'pit'	
6	1.048 24.580	10.895	3.917	-0.434	'pit'	

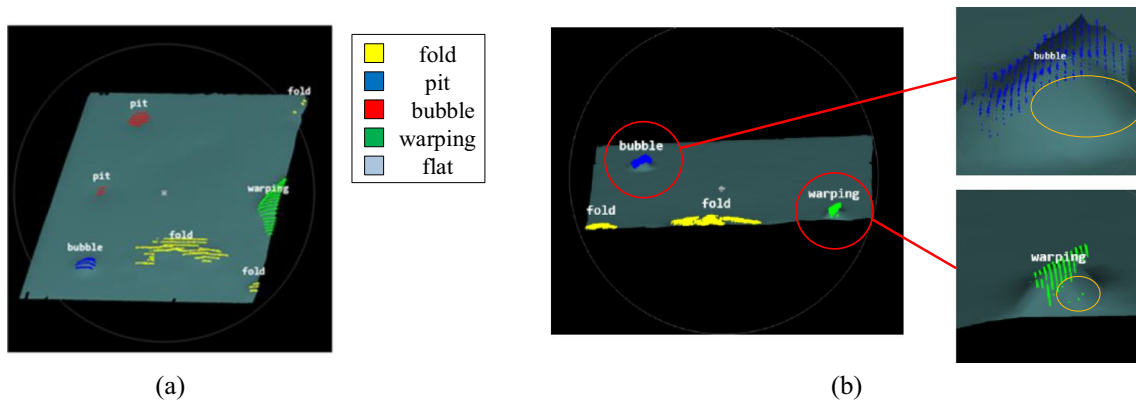


Fig. 13 The classification results for different objects in the database. **a** The result of classification. **b** The missing parts of the model data.

4.4 Evaluation for defect recognition

This section evaluates the full methodology of surface defect recognition proposed in this paper. The set of 128 test depth images consisted of 242 regions labeled as folds, bubble, pit, and warping.

In the detection stage, the points of the surface are classified into five primitives, and the points belonging to a fold (yellow), bubble (blue), pit (red), warping (green), and flat (cadet blue) surfaces. The results of classifying the regions of surface determined by the property of defect can be seen in Fig. 13a.

For robustness testing, Fig. 13 shows the classification results for different objects in the database; our system could be able to classify these defects correctly from the recognition stage of the image. However, it can be seen that some special limitations such as the edge of the warping defect in the acquisition process due to excessive curvature caused by the lack of point cloud will be considered as a fold, and there is a vertical part between the bubble area and the upturned area, which leads to the loss of some point cloud data; thus, the height of the measured defect could be less than the actual defect height, which as a reason affects the accuracy of the identification. But since the other parts of the defect point cloud data are relatively smooth, some height features can be detected, so there is less impact on the detection results. The missing parts of the model data are circled in yellow as shown in Fig. 13b.

The experiment was carried out as follows: 16 experimental lithium batteries were tested and 128 depth images were generated by rotation and inversion operations. The experimental results of 128 images for surface defects detection of lithium are shown in Table 6, which illustrates that there are two false positives in the process of detecting 242 defects. The false detection rate is 0.8%, and the correct detection rate is 99.2%. The accuracy of visual detection is very high, and the efficiency is greatly improved compared with manual detection.

The average time consumption of the lithium battery automatic detection system shown in Table 7 was 3.2 ms for data acquisition, 35.3 ms for the data segmentation step, and 15.5 ms for the classification step. In summary, the automatic detection system could complete the surface defect detection of lithium batteries in 54 ms.

In order to further verify the superiority of the algorithm proposed in this paper, we find three latest surface defect detection methods for comparison with our proposed method. The comparison parameters include running time, accuracy rate, recall rate, and F1-score, and the comparison results are shown in Table 8. It can be seen from the results that the algorithm proposed in this paper is superior to other algorithms in terms of detection accuracy, recall rate, and F1 score. In order to ensure the safety and stability of lithium battery, accuracy is the most priority index parameter, and the running time of the algorithm in this paper can fully meet the industrial demand. Moreover, the algorithm we propose

Table 6 Experimental result of surface defect detection of lithium batteries

Number of images	Mark defects	Number of false positives	Accuracy
128	242	2	99.2%

Table 7 The system time consumption

Step	Data acquisition	Data segmentation	Classification	Sum
Time consumption (ms)	3.2	35.3	15.5	54

Table 8. Defect detection method comparison results

Model	Time consumption (ms)	Accuracy (%)	Recall (%)	F1-score (%)
BSR-Snet [31]	60.11	93.33	88	90
Improved YOLOv3 [32]	13.9	93.63	89.41	90.39
PSO-SVM [9]	-	98.3	100	99.1
The proposed method.	54	99.2	100	99.6

does not need to be based on large datasets. Therefore, the improved algorithm in this paper is excellent.

5 Application of the proposed approach

In this section, a system based on proposed approach had been developed and applied in the field of industrial production as shown in Fig. 14.

Shown in Fig. 14 is the use of computer terminals to control equipment and adjust parameters for defect detection during lithium battery industrial production. Based on the method presented in this paper, the system is used to detect the surface defects of lithium battery and display them in real time. The software system can also store and record the information of lithium battery in real time, which is convenient for historical information inquiry. When defects are found that do not meet the actual

requirements, an alert will be issued for further processing. Our proposed defect detection system has been running stably in the factory for 20 days, and the accuracy of defect detection has reached 99.8%. The application results show that the surface defect detection system of lithium battery can accurately construct the three-dimensional model of lithium battery surface and identify the defects on the model, improving the production quality and efficiency of lithium battery.

6 Conclusions and future work

In this paper, we propose an efficient and accurate method for the real-time detection of high characteristic defects on the surface of lithium batteries. The 3D structured light is used to obtain the surface defects data. The sparse point filtering and the point cloud under-segmentation are solved

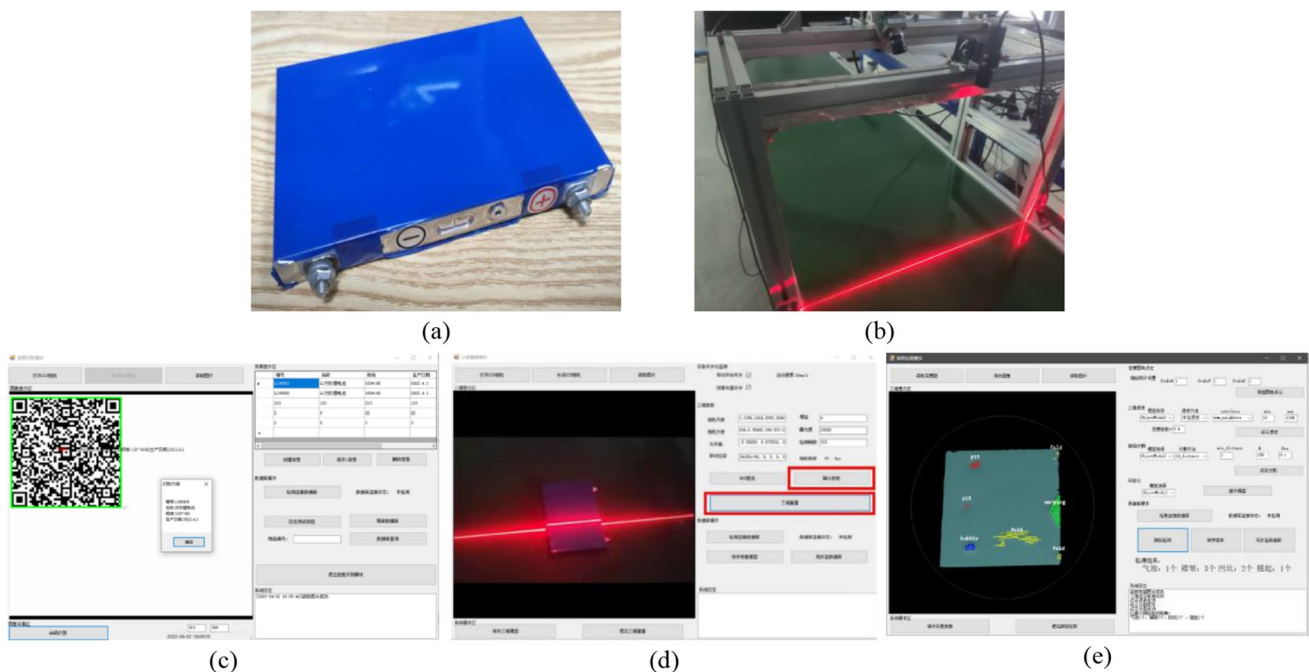


Fig. 14 Industrial application example of proposed method: **a** the object to be detected, **b** detection bench; **c** information record in software interface; **d** the process of 3D reconstruction; **e** the result of defect detection

by voxel-based KNN and improved European algorithm. In this process, we introduce the voxel density ρ as the filtering parameter and the discreteness S as the parameter of Euclidean defect segmentation. Using this method and 2D features obtained by fitting 3D defect points, we quantitatively and qualitatively classify the region into four defects, and the detection accuracy is up to 99.20%. The model we have proposed exhibits high computational complexity, indicating the possibility of further optimizing its operational efficiency. In the future work, we will automatically select detection parameters based on machine learning and combine 2D vision to identify the defects without high-level features in order to further improve our lithium battery detection system.

Acknowledgements The support of National Natural Science Foundation of China (No. 51975568), Natural Science Foundation of Jiangsu Province (No. BK20191341), Jiangsu Funding Program for Excellent Postdoctoral Talent (2022ZB519) and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) are gratefully acknowledged.

Availability of data and materials All data and materials used to produce the results in this article can be obtained upon request from the corresponding authors.

Author contribution Conceptualization: Xinhua Liu and Zhixiong Li. Methodology: Lequn Wu and Xiaoqiang Guo. Software: Xiaoqiang Guo and Darius Andriukaitis. Writing—review and editing: Lequn Wu, Grzegorz Królczyk and Zhixiong Li. Funding acquisition: Xinhua Liu.

Funding The research leading to these results has received funding from the Norwegian Financial Mechanism 2014–2021 under Project Contract No 2020/37/K/ST8/02748.

Declarations

Ethical approval The authors declare that there is no ethical issue applied to this article.

Consent to participate The authors declare that all authors have read and approved to submit this manuscript to IJAMT.

Consent to publish The authors declare that all authors agree to sign the Transfer of Copyright for the Publisher to publish this article upon acceptance.

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Karimzadeh S, Safaei B, Jen T-C (2023) Investigation on electrochemical performance of striped, beta12 and chi3 borophene as anode materials for lithium-ion batteries. *J Mol Graph Model* 120:108423–108423
- Solyali D, Safaei B, Zargar O et al (2022) A comprehensive state-of-the-art review of electrochemical battery storage systems for power grids. *Int J Energy Res* 46(13):17786–17812
- İnada AA, Arman S, Safaei B (2022) A novel review on the efficiency of nanomaterials for solar energy storage systems. *J Energy Storage* 55
- Kalaf O, Solyali D, Asmael M et al (2020) Experimental and simulation study of liquid coolant battery thermal management system for electric vehicles: a review. *Int J Energy Res* 45(5):6495–6517
- Chen G, Zhu Xi F, Xu Qing Q et al (2013) Film defects of lithium battery recognition based on brightness and one-against-all support vector machine. *International Conference on Mechatronics and Information Technology (ICMIT 2013)*, pp 155–158
- Badmos O, Kopp A, Bernthaler T et al (2020) Image-based defect detection in lithium-ion battery electrode using convolutional neural networks. *J Intell Manuf* 31(4):885–897
- Chen Y, Shu Y, Li X et al (2021) Research on detection algorithm of lithium battery surface defects based on embedded machine vision. *J Intell Fuzzy Syst* 41(3):4327–4335
- Lang X, Zhang Y, Shu S et al (2021) Lithium battery surface defect detection based on the YOLOv3 detection algorithm. *10th International Symposium on Precision Mechanical Measurements*
- Xu C, Li L, Li J et al (2021) Surface defects detection and identification of lithium battery pole piece based on multi-feature fusion and PSO-SVM. *Ieee Access* 9:85232–85239
- Yang M, Mo Y (2021) Interfacial defect of lithium metal in solid-state batteries. *Angew Chem, Int Ed* 60(39):21494–21501
- Bullejos M, Cabezas D, Martin-Martin M et al (2022) A K-nearest neighbors algorithm in Python for visualizing the 3D stratigraphic architecture of the Llobregat River Delta in NE Spain. *J Mar Sci Eng* 10(7)
- Liu H, Song R, Zhang X et al (2021) Point cloud segmentation based on Euclidean clustering and multi-plane extraction in rugged field. *Meas Sci Technol* 32(9)
- Wang X, Wu S, Liu Y (2017) Detecting wood surface defects with fusion algorithm of visual saliency and local threshold segmentation. *9th International Conference on Graphic and Image Processing (ICGIP)*
- Ma J, Wang Y, Shi C et al (2018) Fast surface defect detection using improved Gabor filters. *25th IEEE International Conference on Image Processing (ICIP)*, pp 1508–1512
- Moradi N, Kandi SG, Yahyaei H (2022) A new approach for detecting and grading blistering defect of coatings using a machine vision system. *Measurement* 203
- Liu Y, Chen Y, Xu J et al (2020) An automatic defects detection scheme for lithium-ion battery electrode surface. *International Symposium on Autonomous Systems (ISAS)*, pp 94–99
- Liu Z, Yang B, Duan G et al (2020) Visual defect inspection of metal part surface via deformable convolution and concatenate feature pyramid neural networks. *IEEE Trans Instrum Meas* 69(12):9681–9694
- Hao R, Lu B, Cheng Y et al (2021) A steel surface defect inspection approach towards smart industrial monitoring. *J Intell Manuf* 32(7):1833–1843
- Zhao L, Li F, Zhang Y et al (2020) A deep-learning-based 3D defect quantitative inspection system in CC products surface. *Sensors* 20(4)
- Yan Z, Shi B, Sun L et al (2020) Surface defect detection of aluminum alloy welds with 3D depth image and 2D gray image. *Int J Adv Manuf Technol* 110(3–4):741–752

21. Madrigal CA, Branch JW, Restrepo A et al (2017) A method for automatic surface inspection using a model-based 3D descriptor. *Sensors* 17(10)
22. Qian J, Feng S, Xu M et al (2021) High-resolution real-time 360 degrees 3D surface defect inspection with fringe projection profilometry. *Opt Lasers Eng* 137
23. Yan J (2022) Noncontact defect detection method of automobile cylinder block based on SVM algorithm. *Mob Inf Syst* 2022
24. Chen H, Zhang Z, Yin W et al (2022) Surface defect characterization and depth identification of CFRP material by laser line scanning. *NDT E Int*:130
25. Chen Z, Deng J, Zhu Q et al (2022) A systematic review of machine-vision-based leather surface defect inspection. *Electronics* 11(15)
26. Wang J, Wang C, Xi X et al (2022) Segmentation of the communication tower and its accessory equipment based on geometrical shape context from 3D point cloud. *Int J Digit Earth* 15(1):1547–1566
27. Dhanachandra N, Chanu YJ (2018) A new image segmentation method using clustering and region merging techniques. *International Conference on Signals, Machines and Automation (SIGMA)*, pp 603–614
28. Lei X, Ouyang H (2019) Image segmentation algorithm based on improved fuzzy clustering. *Cluster Comput-J Netw Softw Tools Applic* 22:13911–13921
29. Zhou L, Wei Y (2020) DIC: deep image clustering for unsupervised image segmentation. *Ieee Access* 8:34481–34491
30. Liu H, Zhao F (2021) Multiobjective fuzzy clustering with multiple spatial information for noisy color image segmentation. *Appl Intell* 51(8):5280–5298
31. Wu K, Tan J, Liu C (2022) Cross-domain few-shot learning approach for lithium-ion battery surface defects classification using an improved Siamese network. *Ieee Sens J* 22(12):11847–11856
32. Shu YF, Li B, Li X et al (2021) Deep learning-based fast recognition of commutator surface defects. *Measurement* 178

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.