Juan Du
Mem. ASME
Smart Manufacturing Thrust, Systems Hub,
The Hong Kong University of Science
and Technology;
Guangzhou HKUST Fok Ying Tung Research
Institute,
Guangzhou 511458, China
e-mail: juandu@ust.hk

Hao Yan[1]
School of Computing, Informatics, & Decision
Systems Engineering,
Arizona State University,
Tempe, AZ 85281
e-mail: haoyan@asu.edu

Tzyy-Shuh Chang
OG Technologies,
Ann Arbor, MI 48108
e-mail: chang@ogtechnologies.com

Jianjun Shi
Fellow ASME
H. Milton Stewart School of Industrial
and Systems Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332
e-mail: jianjun.shi@isye.gatech.edu

A Tensor Voting-Based Surface Anomaly Classification Approach by Using 3D Point Cloud Data

Advanced three-dimensional (3D) scanning technology has been widely used in many industries to collect the massive point cloud data of artifacts for part dimension measurement and shape analysis. Though point cloud data has product surface quality information, it is challenging to conduct effective surface anomaly classification due to the complex data representation, high-dimensionality, and inconsistent size of the 3D point cloud data within each sample. To deal with these challenges, this paper proposes a tensor voting-based approach for anomaly classification of artifact surfaces. A case study based on 3D scanned data obtained from a manufacturing plant shows the effectiveness of the proposed method. [DOI: 10.1115/1.4052660]

Keywords: surface monitoring, anomaly classification, tensor voting, 3D point cloud data, product surface inspection, inspection and quality control, metrology, sensing, monitoring, and diagnostics

1 Introduction

The forming process, such as stamping, forging, or rolling, is a complex manufacturing process where anomalies may occur on surfaces of manufactured artifacts. Here, we define the anomaly as the quality anomaly on manufactured artifacts that is anything unacceptable by quality standards or by industrial practices. Many types of anomalies can be generated in a forming process at different stages [1]. Different anomalies may have different potential root causes, and classification of surface anomalies helps diagnosis of root causes. Thus, surface anomaly classification is very critical in a forming process for better decision-making to address anomaly or defective products. In addition, modeling and analysis of each class of anomalies can help understand the defect formation and develop actions to reduce or eliminate the anomalies. Therefore, the ability to successfully perform surface anomaly classification for fault diagnosis is one of the key tasks to improve the manufacturing process.

The rapid development of three-dimensional (3D) measurement technologies has created unprecedented opportunities for surface anomaly classification in manufacturing processes. Those advanced 3D scan or metrology systems generate massive data from product surface measurements and are typically referred to as 3D point cloud data. Contrasted to image data, 3D point cloud data, with the extra dimension of information, is able to provide the entire external surface geometry of the manufactured artifacts, making it a promising tool for surface anomaly classification. Figure 1 shows three scanning examples of surface anomalies on steel products, including a depression, a pinhole, and a debris patch. As shown in Fig. 1, 3D point cloud data exhibit different patterns for different surface anomalies. For example, compared with pinhole and debris patch, there are a relatively larger number of points that deviate from the original surface with a significant coordinate shift in depression, which are within the dash line in Fig. 1(*a*). Therefore, 3D point cloud data have the potential for effectively identifying the anomalies on surfaces of manufactured artifacts.

Depending on the 3D metrology system, 3D point cloud data can be further classified into two categories: structured point cloud data and unstructured point cloud data. Structured point cloud data can be collected from coordinate measuring machines (CMMs) [2,3] or structure highlight scanners [4], where points are measured on a pre-specified grid. Thus, structured point cloud data can be treated as matrix data or tensor data that can be further modeled by the tensor decomposition or tensor regression methods [5,6]. Unstructured point cloud data with coordinates at random measurement locations can be collected from laser scanners, which have become more and more popular in many 3D scanning systems. Compared with CMMs with partial measurements, laser scanners can capture the entire geometry of an artifact. Additionally, the manufacturing industries begin to use laser scanners for surface inspection, where millions of data points on random coordinates are collected to represent the entire surface of the artifacts [7]. As a result, the data representation of unstructured 3D point cloud data is complicated, and the dimensionality of 3D point cloud data is high. The goal of this paper is to propose an effective method for anomaly classification on surfaces of manufactured artifacts by using unstructured point cloud data.

Although there is extensive research on surface anomaly detection or classification by using images, limited research exists on anomaly classification using point cloud data in the literature. Furthermore, there are significant differences in terms of data characteristics between point cloud data and image data. Image data are well structured and can be easily represented. However, unstructured point cloud data have the following four key characteristics: (1) *Complicated representation*: compared with the structured point cloud (with $\mu\nu\omega$ coordinates) that the measurement grid is predefined (usually specify $\mu\nu$), the points in the unstructured point cloud are not measured at a pre-defined grid, which is often

[1]Corresponding author.

Manuscript received April 7, 2021; final manuscript received September 15, 2021; published online October 25, 2021. Assoc. Editor: Qiang Huang.

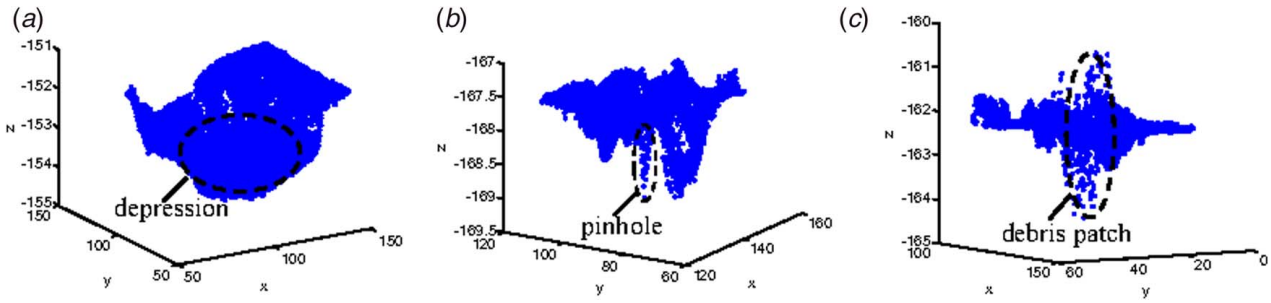

Fig. 1 Illustrations of three types of anomalies on steel surfaces: (*a*) depression, (*b*) pinhole, and (*c*) debris patch

challenging to represent effectively in a data matrix that is commonly used for other machine learning applications. Here, one sample is one point cloud. (2) *High-dimensionality*: one typical point cloud sample often has millions of data points, posing a significant challenge for statistical modeling. We call the number of points within one point cloud as point size. (3) *Unequal point sizes within different point cloud samples*: the number of measurement points is often different from one point cloud sample to another. For example, in our case study, this number ranges from 3564 to 96,266, which is challenging to extract the same dimension of features. (4) *Localized sparse anomaly*: the surface anomaly points usually form a localized, contiguous group in the entire 3D point cloud data. Compared with the normal points, most anomaly cases are with a small number of anomaly points, which is sparse. Thus, the development of an approach that can distinguish different anomaly samples is challenging. In short, those four data characteristics make the development of the surface anomaly classification method by using unstructured 3D point cloud data extremely challenging.

To address the aforementioned challenges of anomaly classification based on the unstructured point cloud, this paper proposes a tensor voting-based approach to analyze the 3D point cloud data directly for surface anomaly classification. The framework of the proposed methodology is shown in Fig. 2. Tensor voting is able to extract useful local geometry information by aggregation of the spatial information from the neighborhood, which helps localize the surface anomaly points. Thus, tensor voting has been used to characterize the local geometry of artifact surfaces. Based on the tensor voting method, a surface region can be segmented into different sub-regions and several points that represent the local anomaly can be selected for feature extraction. To deal with the challenges of 3D data representation and unequal point sizes among point cloud data samples, we propose to use descriptive statistics to extract useful features from the selected points by considering the domain knowledge from the forming process. Finally, the multiclass sparse support vector machine (SVM) classifier is applied for anomaly classifications on the artifact surfaces.

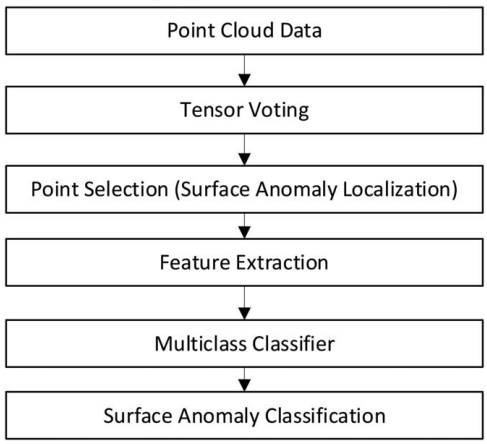

Fig. 2 The framework of the tensor voting-based surface anomaly classification approach

The key contributions of the paper are as follows: (1) propose a systematic framework to deal with the unstructured 3D point cloud data directly; (2) develop a point selection method to localize the anomaly; and (3) propose a descriptive feature extraction method to deal with the unequal point size within each point cloud sample for further classification.

The remainder of this paper is organized as follows. We review the related work in Sec. 2. Section 3 provides a basic introduction to tensor voting. Then, Sec. 4 presents the proposed tensor voting-based approach for anomaly classification of the artifact surfaces. A real case study using data from a steel plant is provided to exemplify the performance of our method in Sec. 5. Finally, Sec. 6 concludes this paper.

2 Literature Review

This section reviews the current literature on monitoring and classification using 3D point cloud data. Due to the different data characteristics between structured and unstructured 3D point cloud data, the methodologies developed for process modeling, monitoring, and classification by using these two types of data differ in the literature. We review the related literature based on this perspective and classify the literature into the following two categories: structured and unstructured 3D point cloud data, with an emphasis on the literature review by using unstructured 3D point cloud data.

The *first* category mainly deals with *structured point cloud data*. For example, a spatial autoregressive regression model-based procedure, which aims to represent both the large-scale and small-scale spatial correlation characteristics of flat cylindrical surfaces, is proposed for cylindrical surface monitoring [8]. To consider the 3D curved surface monitoring, Huang et al. [9] proposed a curved surface monitoring procedure that includes region division, feature evaluation, and quality parameter calculation and monitoring. However, this method does not consider the nonstationary relationship between surface covariate and surface height. Then, a transfer learning framework is developed to fuse multi-resolution data for surface variation modeling with consideration of the nonstationary relationship [10]. However, in their work, most surface data are low resolution. To analyze the high-resolution 3D point cloud data, a regularized tensor regression method is proposed to quantify the relationship between structured point clouds and scalar predictors with consideration of dimension reduction in Ref. [5]. However, the aforementioned methods cannot be applied for the unstructured point cloud data due to their random measurement grids of point coordinates.

The *second* category mainly deals with *unstructured point cloud data*. For example, Wells et al. [7] proposed to use the linear profile monitoring technique to deal with the unstructured point cloud data

by transforming the point cloud data to the distribution of deviations from the nominal surface. Based on this work, Wells et al. [11] further proposed an adaptive generalized likelihood ratio (AGLR) technique to automate the surface anomaly inspection by transforming the 3D point cloud data to a 2D matrix. Dastoorian et al. [4] then extended this work by relaxing the assumption of identically distributed deviations and tested the AGLR method by a real case study. All these three methods aimed to monitor the process by using an unstructured point cloud via transforming the 3D point cloud to a profile or 2D image data, thereby resulting in the loss of the advantageous information of the extra dimension of the 3D point cloud data. Recently, Zhao and Del Castillo [12] also proposed an intrinsic geometrical approach for process monitoring by using the surface and manifold data. They need in-control dataset for setting up the in-control limits and cannot identify the surface anomaly type.

In addition to process monitoring, there are also research works focusing on classification by using unstructured point cloud data in manufacturing applications. For example, targeting on the dimensional variation classification of additive manufactured parts, Samie Tootooni et al. [13] developed a spectral-graph based approach by using the deviations of 3D cloud data from computer-aided design models of the corresponding part. Yacob et al. [14] proposed the use of the histogram of deviations to represent the 3D point cloud data of skin model shapes from the nominal model and classified the random and systematic deviations using the histogram features for anomaly detection of manufactured artifacts. Nonetheless, all these methods assume the nominal model is known so that the deviations can be calculated.

Since this paper aims at surface anomaly classification by using unstructured point cloud data, we would like to briefly introduce other literature on the classification problem by using unstructured point cloud data in the field of computer vision. The current research literature in computer vision mainly focuses on 3D object classification, which classifies the object category from 3D data. For example, Himmelsbach et al. [15] proposed using feature histograms to realize real-time object classification in 3D point clouds. Wohlkinger and Vincze [16] proposed an ensemble of shape functions for 3D shape-based object class recognition by using a partial point cloud surface. Recently, deep learning-based methods are also popular for point cloud data analysis. For example, Qi et al. [17] proposed to use a deep learning-based model (called PointNet) for 3D cloud data classification and segmentation. All these classification methods focus on the global object classification, such as "mug," "table," and "car" [17], or the object classification/recognition within one point cloud based on the segmentation [18]. Based on this work, Qi et al. [19] further proposed PointNet++ to learn local features, and Zhao et al. [20] proposed PointWeb to enhance the neighborhood features based on PointNet++. However, deep learning-based features are usually challenging to interpret and require a large sample size to learn good features. But the sample size with anomalies is relatively small in manufacturing applications, which cannot meet the sample size requirements of deep learning methods.

In summary, there is very limited work on surface anomaly classification in dealing with unstructured point cloud data. This paper aims to fill these research gaps and proposes a method for surface anomaly classification by using unstructured point cloud data.

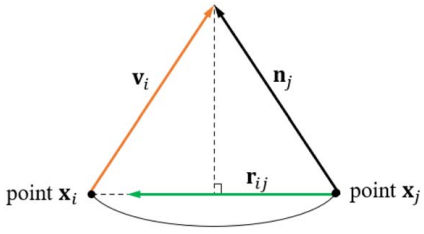

Fig. 3 Illustration of the normal vote $\mathbf{v}_i$ received at $\mathbf{x}_i$ from $\mathbf{x}_j$

3 Introduction to Tensor Voting

To better illustrate the proposed tensor voting-based approach, we first provide a basic introduction to tensor voting in this section. Tensor voting, which makes inference on the geometry information such as surface, curve, and junction via a voting scheme over the neighborhood, is originated from Guy and Medioni [21]. Based on this work, the tensor voting method is formally proposed in Ref. [22], in which a unified computational framework is illustrated to extract the geometric descriptions from 2D and 3D data. The tensor voting approach mainly depends on tensor calculus for point data representation and the voting scheme for point data communication.

In tensor voting, for each data point $\mathbf{x}_i \in R^3$, a structure-aware tensor $\boldsymbol{K}_i$ is encoded via the vote collection from its neighborhood $N(\mathbf{x}_i)$. Figure 3 shows a simple case with only two points $\mathbf{x}_i$ and $\mathbf{x}_j$ exist. We will start with a special case where the unit norm vector $\mathbf{n}_j$ at $\mathbf{x}_j$ is known. $\mathbf{v}_i$ is the normal vote received at $\mathbf{x}_i$ by using osculating arc connection; $\mathbf{r}_{ij}$ is a unit vector at $\mathbf{x}_j$ pointing to $\mathbf{x}_i$. Then, the second-order symmetric tensor vote $\boldsymbol{S}_{ij}$ is then given by $\mathbf{v}_i\mathbf{v}_i^T$ multiplied by $\eta(\mathbf{x}_i, \mathbf{x}_j, \mathbf{n}_j)$, defined as

$$\eta(\mathbf{x}_i, \mathbf{x}_j, \mathbf{n}_j) = c_{ij}(1 - (\mathbf{r}_{ij}^T\mathbf{n}_j)^2),\ c_{ij} = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma_d) \quad (1)$$

where σ_d is a scale parameter; $\mathbf{r}_{ij}$ is a unit vector at $\mathbf{x}_j$ pointing to $\mathbf{x}_i$. The unit direction $\mathbf{v}_i$ can be derived by fitting an arc of the osculating circle between the two points.

Then, we will look at a general case, where the unit normal $\mathbf{n}_j$ at $\mathbf{x}_j$ is not known. Let $\boldsymbol{K}_j$ is any second-order symmetric tensor at $\mathbf{x}_j$, which is typically initialized as an identity matrix if $\mathbf{n}_j$ is not available. Then all possible unit normals $\{\mathbf{n}_{\theta j}\}$ with the corresponding length $\{\tau_{\theta j}\}$ are considered with all possible directions θ. Then, the tensor vote $\boldsymbol{S}_{ij}$ for $\mathbf{x}_i$ from $\mathbf{x}_j$ can be obtained as

$$\boldsymbol{S}_{ij} = \int_{\mathbf{N}_{\theta j} \in v} \mathbf{v}_\theta(\mathbf{x}_i, \mathbf{x}_j)\mathbf{v}_\theta(\mathbf{x}_i, \mathbf{x}_j)^T \eta(\mathbf{x}_i, \mathbf{x}_j, \mathbf{n}_{\theta j}) d\mathbf{N}_{\theta j} \quad (2)$$

where $\mathbf{N}_{\theta j} = \mathbf{n}_{\theta j}\mathbf{n}_{\theta j}^T$, and v is the space that contains all possible $\mathbf{N}_{\theta j}$. By using the osculating arc connection, $\mathbf{v}_\theta(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{n}_{\theta j} - 2\mathbf{r}_{ij}(\mathbf{r}_{ij}^T\mathbf{n}_{\theta j}))\tau_{\theta j}$. The integral is typically numerically computed by the voting field [22] until Wu et al. [23] proposed the closed-form solution in 2012, which makes the tensor voting method easier to calculate and implement.

In this paper, the closed-form solution in Ref. [23] is briefly introduced and adopted to perform tensor voting. The structure-aware tensor $\boldsymbol{K}_i$ at point $\mathbf{x}_i$ can be calculated as

$$\boldsymbol{K}_i = \sum_j \boldsymbol{S}_{ij},\ \boldsymbol{S}_{ij} = c_{ij}\boldsymbol{R}_{ij}\boldsymbol{K}_j\left(\mathbf{I} - \frac{1}{2}\mathbf{r}_{ij}\mathbf{r}_{ij}^T\right)\boldsymbol{R}_{ij}^T \quad (3)$$

where $\boldsymbol{S}_{ij}$ is the tensor vote cast by $\mathbf{x}_j$ with respect to $\mathbf{x}_i$; $\boldsymbol{R}_{ij} = \mathbf{I} - 2\mathbf{r}_{ij}\mathbf{r}_{ij}^T$, where $\mathbf{I}$ is the identity matrix; $\boldsymbol{r}_{ij} = (\mathbf{x}_i - \mathbf{x}_j)/\|\mathbf{x}_i - \mathbf{x}_j\|$, i.e., a unit vector pointing from $\mathbf{x}_j$ to $\mathbf{x}_i$; $\mathbf{x}_j$ is in the neighborhood of $\mathbf{x}_i$, i.e., $\mathbf{x}_j \in \mathrm{N}(\mathbf{x}_i)$; $\boldsymbol{K}_j$ encodes the information of the local geometry around point $\mathbf{x}_j$.

The resulting structure-aware tensor $\boldsymbol{K}_i$ can be visualized as an ellipsoid, where the shape of the ellipsoid provides local geometrical information carried in a 3D point cloud. Generally, a second-order symmetric tensor can be represented by the corresponding three eigenvectors and eigenvalues via spectral decomposition [24], which is shown as follows:

$$\boldsymbol{K}_i = (\lambda_1 - \lambda_2)\boldsymbol{T}_{stick} + (\lambda_2 - \lambda_3)\boldsymbol{T}_{plate} + \lambda_3\boldsymbol{T}_{ball} \quad (4)$$

where $\boldsymbol{T}_{stick}$, $\boldsymbol{T}_{plate}$, and $\boldsymbol{T}_{ball}$ are stick tensor, plate tensor, and ball tensor, respectively; λ_1, λ_2, and λ_3 ($\lambda_1 \geq \lambda_2 \geq \lambda_3$) are the eigenvalues associated with eigenvectors $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ of a second-order

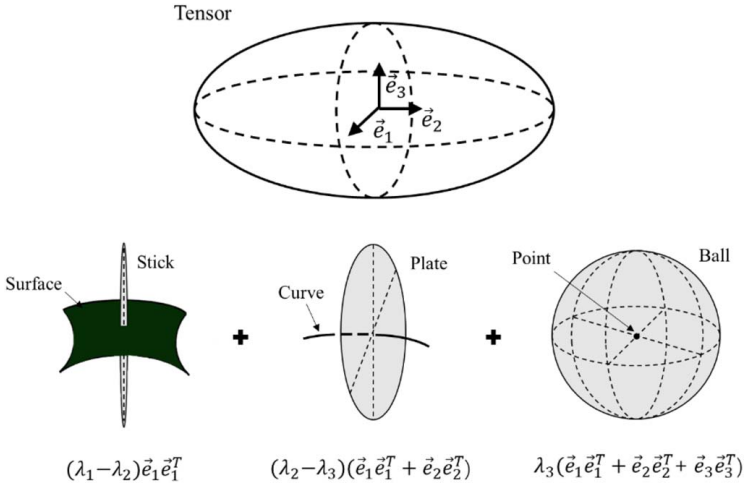

Fig. 4 Illustration of 3D generic tensor decomposition

symmetric tensor $\boldsymbol{K}_i$; and $\boldsymbol{T}_{stick}$, $\boldsymbol{T}_{plate}$, $\boldsymbol{T}_{ball}$ can be further represented as

$$\boldsymbol{T}_{stick} = \mathbf{e}_1\mathbf{e}_1^T,\ \boldsymbol{T}_{plate} = \mathbf{e}_1\mathbf{e}_1^T + \mathbf{e}_2\mathbf{e}_2^T,\ \boldsymbol{T}_{ball} = \mathbf{e}_1\mathbf{e}_1^T + \mathbf{e}_2\mathbf{e}_2^T + \mathbf{e}_3\mathbf{e}_3^T \tag{5}$$

Notably, such a decomposition also validates on generic tensor according to spectral decomposition [24]. Figure 4 shows a generic tensor that can be decomposed into the stick, plate, and ball components.

In tensor voting, the decomposed eigenvalues and eigenvectors from the structure-aware tensor $\boldsymbol{K}_i$ are used to extract the geometric information of a surface, curve, or junction. $\lambda_1-\lambda_2$, $\lambda_2-\lambda_3$, and λ_3 are the surface, curve, and junction saliency, which characterizes the local geometric information according to stick, plate, and ball components, respectively. The local structure inference can be obtained from the maximizer of these three saliency features. For example, if $\lambda_1-\lambda_2$ is the largest, i.e., $\lambda_1-\lambda_2 > \lambda_2-\lambda_3$, and $\lambda_2-\lambda_3 > \lambda_3$, the local structure of the corresponding point is more likely to be part of a surface (i.e., surface-ness). On the other hand, if $\lambda_2-\lambda_3$ is the largest, it indicates the local structure of the corresponding point is more likely being part of a curve (i.e., curve-ness).

Figure 5 further illustrates the inference of local structures and related dominant tensor components. The black dots are the points that are the neighbors around the red point within one point cloud. As shown in Fig. 5(*a*), the resulting tensor ellipsoid is more like a stick with a direction of the surface normal if the point is on a smooth surface. If a point belongs to a curve, the resulting tensor ellipsoid is more like a plate, shown in Fig. 5(*b*). Finally, if a point is a surface intersection point, i.e., junction point, the resulting tensor ellipsoid will be more like a ball, shown in Fig. 5(*c*). With such inferences, the application of the tensor voting method provides valuable information about the local geometry on artifact surfaces to facilitate anomaly classification. Finally, $\mathbf{e}_1$ denotes the surface normal or direction of the stick component, $\mathbf{e}_1$, $\mathbf{e}_2$ denote the direction of the curve component, the ball does not have a specific direction and therefore is spanned on all directions, denoted as $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$. The corresponding interpretations are summarized in Table 1.

4 The Tensor Voting-Based Approach for Surface Anomaly Classification

Based on the tensor voting theory, this paper proposes a novel approach for surface anomaly classification by using unstructured 3D point cloud data. The main goal of this paper is to classify the anomaly type on the artifact surface by using the unstructured 3D point cloud data obtained from the artifact even though the data have inconsistent point sizes. Notably, this approach directly deals with 3D point cloud data without transforming the 3D representation to 2D.

To begin with, we will first introduce our notations and the problem setup. The problem to be solved is to predict the type of anomaly carried in a new point cloud sample given the historical point cloud samples with different types of anomalies. Assume that we have n samples (i.e., n different point cloud sets), and that each sample may contain a different number of measurement points. We will use m_k to denote the number of points in the kth point cloud sample. In addition, each point $\mathbf{x}_i = (\mu_i, \nu_i, \omega_i) \in R^3$ is defined in the three-dimensional μ, ν, ω coordinates. Let M denote the number of neighbors that will be used for local aggregation in the tensor voting process.

Given the problem setup, our overall approach is illustrated in Fig. 6, which consists of the following steps for each point cloud sample: (1) *Tensor voting for saliency feature extraction*: for all the m_k points of the point cloud sample k, tensor voting is first

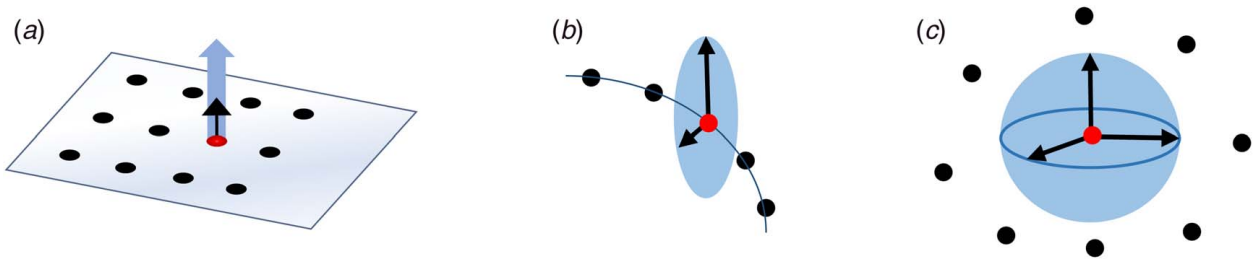

Fig. 5 The illustration of local structures and related dominant tensor components: (*a*) a stick tensor T_{stick} for a surface, (*b*) a plate tensor T_{plate} for a curve, and (c) a ball tensor T_{ball} for a junction

Table 1 The interpretations of structure-aware tensor decomposition around a point

	Saliency	Tensor component	Normal
Surface-ness	$\lambda_1 - \lambda_2$	$\boldsymbol{T}_{stick}$	$\mathbf{e}_1$
Curve-ness	$\lambda_2 - \lambda_3$	$\boldsymbol{T}_{plate}$	$\mathbf{e}_1$, $\mathbf{e}_2$
Point-ness	λ_3	$\boldsymbol{T}_{ball}$	$\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$

applied to extract the geometry saliencies as the point-level features, such as surface saliency $\alpha_i = \lambda_{i,1} - \lambda_{i,2}$, curve saliency $\beta_i = \lambda_{i,2} - \lambda_{i,3}$, and junction saliency $\gamma_i = \lambda_{i,3}$, of each point $\mathbf{x}_i$, i, $i = 1, \ldots, m_k$, where $\lambda_{i,1}$, $\lambda_{i,2}$, and $\lambda_{i,3}$ are the decomposed eigenvalues of structure-aware tensor as introduced in Sec. 2. (2) *Sharp points selection*: given these point-level features, we can achieve the point selection according to saliency segmentation. More specifically, we aim to select important points that include potential anomalies based on the geometry information in the point cloud sample. These selected points are defined as the "sharp points." (3) *Descriptive features aggregation*: since the number of sharp points can be different for each point cloud sample, we propose to use the aggregated descriptive features of each point cloud sample for further surface anomaly classification. The aggregated descriptive features will be of the same size for all point cloud samples. (4) *Anomaly classification*: These aggregated features will be passed to the sparse multiclass SVM classifier for surface anomaly classification and feature selection. These steps will be introduced in Secs. 4.1–4.3. Notably, we do not need any registrations between different point cloud samples since we deal with each point cloud sample independently.

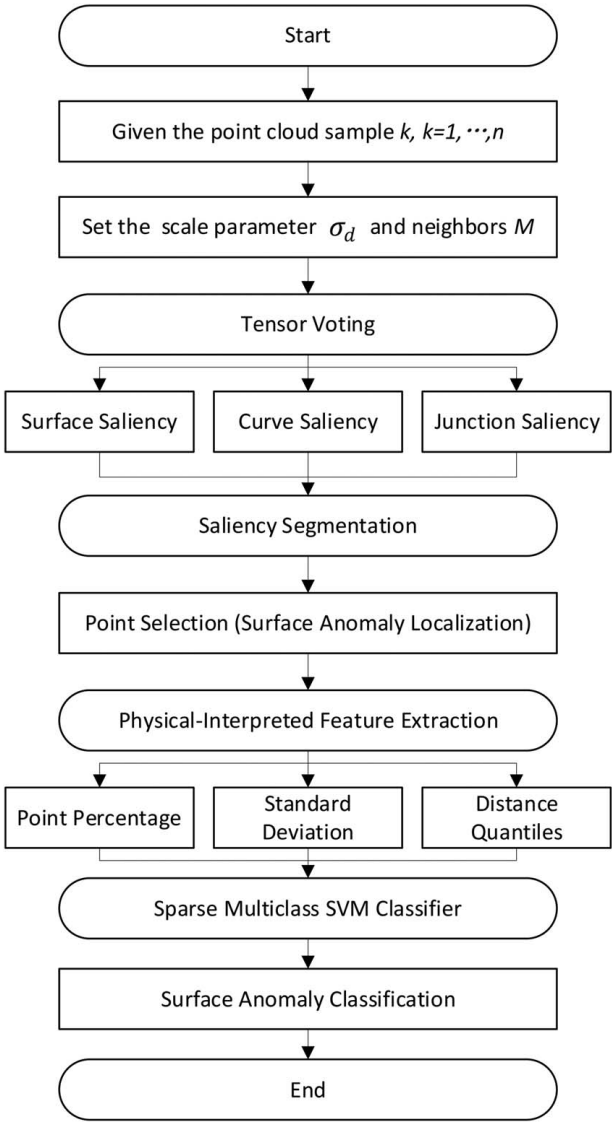

Fig. 6 Flowchart of the proposed approach

4.1 Saliency Segmentation. Notably, the points with similar saliency values may have similar local information, which will facilitate further feature extraction. Thus, we group the points according to the saliency values. One simple way is to first sort the saliencies and then segment the sorted saliency values to group points. Specifically, for the kth point cloud sample, we first sort each type of the saliencies α_i, β_i, γ_i associated with each point, $\mathbf{x}_i$, i, $i = 1, \ldots, m_k$, respectively. Let $\alpha_{(i)}$ denote the ith sorted surface saliency, i.e., $\alpha_{(1)} \geq \alpha_{(2)} \geq \ldots \geq \alpha_{(m_k)}$. Similarly, $\beta_{(i)}$ and $\gamma_{(i)}$ denote the sorted curve and junction saliencies, respectively. The segmentation is then desired to segment the sorted saliency values with the point index. Here, we use the piecewise-linear spline with q free knots [25] to fit the relationship between the sorted saliencies, i.e., $\alpha_{(i)}$, $\beta_{(i)}$, or $\gamma_{(i)}$, and the point index (i.e., $1,2,\ldots,m_k$), respectively, for simplicity. The fitted locations of the knots provide the natural segmentation points of the saliency feature, and the sharp points can be determined from the first segmentation of the junction saliency along with the overlapping points between the second segments of the junction and surface saliencies. Let $\mathcal{H}_k$ denote the set of sharp points for the kth point cloud sample, and $\mathcal{R}_k$ denote the regular points, i.e., the points in the kth point cloud sample excluding the sharp points.

4.2 Feature Extraction. In general, different types of surface anomalies may have different types of patterns and associated causes [1,26]. For example, the surface anomalies on steel surfaces may have various patterns. Figure 7 shows three different types of steel surface anomalies, i.e., deviations from the supposed flat. Contrasted to depression, a pinhole usually has a very sharp local pattern, which means few sharp points would be the key features for classification. As another example, depression is typically concentrated while the oscillation marks seem to spread on the steel surface. Thus, a large set of distinct features for different types of surface anomalies need to be defined for point cloud classification in a manufacturing process. Furthermore, feature extraction is important since each point cloud sample may be of different point sizes. There is a need for feature generation techniques to create the same number of features given different sizes of surface point cloud data for further classification models. In this section, we will build a set of common features based on the tensor voting methodology for the manufacturing anomaly classification. The key assumption for feature extraction is that the surface can be assumed known, such as a plane. However, the tensor voting method is not restricted to such an assumption, so the proposed framework can also be useful for other cases that the normal surface has the explicit model. Finally, the set of features we defined are suitable for a variety of anomalies, which we believe could also be useful for other surface anomaly classification tasks.

First, we propose to define a set of features as the percentage of sharp points in a point cloud sample with m_k points. Let s_k denote the number of sharp points in the kth point cloud sample. Then

$$f_{1,k} = s_k / m_k \tag{6}$$

where $f_{1,k}$ is the first feature for the kth point cloud sample. In addition, the location distribution of sharp points also indicates different types of surface anomalies.

Fig. 7 Three types of anomalies on the steel surface: (*a*) pinhole, (*b*) depression, and (*c*) oscillation mark

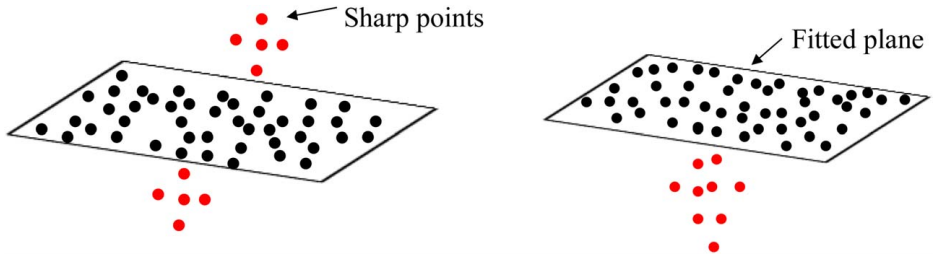

Fig. 8 Illustrations of two surface anomaly patterns: sharp points located on both sides of the plane (left) and sharp points mainly located on one side of the plane (right)

Second, we propose the standard deviations of sharp points as the features, i.e.

$$f_{2,k} = \sigma_\mu(\mathbf{x}_i),\ f_{3,k} = \sigma_\nu(\mathbf{x}_i),\ f_{4,k} = \sigma_\omega(\mathbf{x}_i),\ \mathbf{x}_i \in \mathcal{H}_k \tag{7}$$

where $\sigma_\mu(\cdot)$, $\sigma_\nu(\cdot)$, and $\sigma_\omega(\cdot)$ are the functions of calculating standard deviations in the axial directions of the μ, ν, ω coordinates, respectively. These features are useful for the depression and oscillation mark type of surface anomalies. To avoid the impact on different sizes between different point cloud samples and noises, we also extract the standard deviations of the regular points $\mathcal{R}_k$ as the features to provide more information, i.e.

$$f_{5,k} = \sigma_\mu(\mathbf{x}_i),\ f_{6,k} = \sigma_\nu(\mathbf{x}_i),\ f_{7,k} = \sigma_\omega(\mathbf{x}_i),\ \mathbf{x}_i \in \mathcal{R}_k \tag{8}$$

Notably, although the feature extractions from Eqs. (7) and (8) are based on the absolute coordinates of points in a point cloud, the features we extracted only rely on the relative distance between the points within the point cloud. All the descriptive features (including the percentage of sharp points, standard deviations, or pairwise distance quantiles in the following ((9)–(11)) are rigid transformation-invariant. Thus, there is no registration issue in this paper.

Finally, we will deal with the surface where the original geometry is supposed to be a plane, such as the surfaces in Fig. 1. To further characterize the anomaly that deviates from such surfaces, we first use the points in $\mathcal{R}_k$ to fit a plane given the kth point cloud sample via the singular value decomposition (SVD) based method [27]. Then, the distance d_i to the plane at each point $\mathbf{x}_i$, $\mathbf{x}_i \in \mathcal{R}_k$, can be obtained. The distribution of d_i shows the point distribution around the plane, which helps identify the type of surface anomaly. Thus, we extract quantiles of d_i distribution as features, i.e.,

$$f_{j,k} = F^{-1}_{\mathcal{R}_k}(p_j),\ j = 8,9,\ldots,13 \tag{9}$$

where $F^{-1}_{\mathcal{R}_k}(\cdot)$ is the inverse cumulative distribution function of d_i at point $\mathbf{x}_i$ ($\mathbf{x}_i \in \mathcal{R}_k$), and $f_{j,k}$ is the quantile feature corresponding to the probability p_j. We use the quantile features to fully explore the distance distribution. The detailed quantile values are illustrated in the case study.

To capture different types of anomalies on the surface, the distances of the sharp points to the plane are also considered. As shown in Fig. 8, the pattern with the sharp points located on one side of the plane is different from that of the sharp points located on both sides of the plane for surface anomaly classification. Thus, we distinguish these two cases using the distance calculation. Specifically, we use the angle between the normal vector of the plane and the point vector from the plane to distinguish the sharp points located on either side of the plane, thereby indicating two groups of sharp points. For each group, we can obtain the maximum distance within the group and define the group with the larger maximum distance as the first group, i.e., $\mathcal{H}^1_k$. The other group is denoted as $\mathcal{H}^2_k$. We also calculate the distances to the plane of these two groups. Similarly, quantile features of distances for each group are also extracted for the surface anomaly:

$$f_{j,k} = F^{-1}_{\mathcal{H}^1_k}(p_j)\ ,\ j = 14,15,\ldots,19 \tag{10}$$

$$f_{j,k} = F^{-1}_{\mathcal{H}^2_k}(p_j)\ ,\ j = 20,21,\ldots,25 \tag{11}$$

where $F^{-1}_{\mathcal{H}^1_k}(\cdot)$ and $F^{-1}_{\mathcal{H}^2_k}(\cdot)$ are the inverse cumulative distribution functions of d_i and d_j at point $\mathbf{x}_i$ ($\mathbf{x}_i \in \mathcal{H}^1_k$) and $\mathbf{x}_j$ ($\mathbf{x}_j \in \mathcal{H}^2_k$), respectively; and $f_{j,k}$ is the quantile feature corresponding to the probability p_j.

Given the feature set $f_{i,k}$, $i = 1,\ldots,25$; $k = 1,\ldots,n$, different types of surface anomalies can be classified via a multiclass classifier. We also listed a table to summarize these features, which is shown in Table 2. These features are derived from the manufacturing domain knowledge regarding surface anomalies, so the related physical interpretations can be provided. Next, these extracted features will be used in a classification model to indicate the anomaly types of the artifact surfaces. We would like to mention that additional features may be defined for specific problems. However, we find that these features have worked pretty well for very diverse anomaly types as shown in the case study.

4.3 Multiclass Support Vector Machine Classifier. To identify various types of surface anomalies, a multiclass classifier is

Table 2 Extracted features for the *k*th point cloud sample

Feature	Definition	Physical interpretations
$f_{1,k}$	Percentage of sharp points	Characterization of anomaly size
$f_{2,k} \sim f_{4,k}$	Standard deviations of sharp points	Characterization of anomaly distributions in 3D
$f_{5,k} \sim f_{7,k}$	Standard deviations of the regular points	Characterization of distributions of artifact surface in 3D
$f_{8,k} \sim f_{13,k}$	Quantiles for distance distribution of product surface	Characterization of distance distributions of artifact surface in 3D
$f_{14,k} \sim f_{25,k}$	Quantiles for distance distribution of anomalies along two directions	Characterization of distributions of surface anomalies in 3D

used along with the features extracted in Sec. 4.2. In most manufacturing industries, the sample size with different manufacturing anomalies is usually small. The SVM is initially designed for binary classification problems and shows great success in classification problems with a small sample size [28,29]. Later, the multiclass extension of the binary SVM classifier is also investigated in the literature [30–32]. Therefore, a multiclass SVM classifier is adopted in our paper. Among these methods [30–32], a sparse multiclass SVM classifier [31] is applied for the task of surface defect classification as only a subset of extracted features may be important. The shrinkage idea is used to shrink the unimportant features or redundant features to zero. Specifically, we are using the sup-norm penalty-based sparse multiclass SVM classifier, where the related tuning parameter λ_0 for sup-norm penalty is selected by cross validation. The selection of the sup-norm penalty is reasonable because all the coefficients related to the unimportant feature variable should be all zero. Mathematically, the sup-norm penalty is to penalize the maximum absolute values of K coefficients of each feature variable if K classes exist. More details can be referred to Ref. [31].

5 Case Study

In this section, we present a case study of anomaly classifications on steel product surfaces based on a set of product inspection data obtained from a steel mill. The steel manufacturing process is a complicated manufacturing process, where many types of anomalies may occur due to various factors and interactions of many causes. Surface anomaly classification is very important for steel production to improve the control of the manufacturing process and produce high-quality products. In practice, anomalies on steel products can be of many types, including cracks, depressions, pinholes, etc. Such surface anomalies are local and sparse, posing significant challenges for the classification task in industrial practices.

Table 3 Anomaly types and associated samples

Anomaly class	Anomaly type	Sample size	Point size range
1	Debris patch	30	[6480: 86,784]
2	Oscillation mark	18	[14,317: 63,308]
3	Depression	18	[17,767: 96,266]
4	Slag	92	[3828: 9202]
5	Pinhole	70	[3564: 12,535]

Our proposed approach is validated for this application. We will first introduce the data description in Sec. 5.1 and then show the results of point selections and classification accuracies in Secs. 5.2 and 5.3, respectively. The sensitivity analysis regarding the neighborhood size of tensor voting to the overall approach is illustrated in Sec. 5.4. Comparison studies with the recent existing approach are conducted in Sec. 5.5. Finally, the discussion is provided in Sec. 5.6.

5.1 Basic Data Description. In this case study, we have collected $n = 228$ point cloud samples with five different anomaly types of steel surfaces from a mill. The basic information about each anomaly type and the related point cloud samples is listed in Table 3. Notably, the point size of each point cloud sample varies substantially, ranging from 3564 to 96,266. Table 3 also lists the point size range for each type of anomaly. As shown in Table 3, different anomaly types may have very different point size ranges.

5.2 Result of Tensor Voting and Sharp Point Selection. In our proposed approach of surface anomaly classification, the first step is to use tensor voting for saliency extraction. Related to the parameter σ_d in the tensor voting theory, Wu et al. [33] pointed out that a large range of σ_d, except the extensively large value, would work well. In this case study, we set $\sigma_d = 0.5$, and $M = 300$ for tensor voting. Given these parameters, the surface, curve, and junction saliencies of each point within each point cloud sample can be obtained.

We would like to select a point cloud sample for illustration purposes. Figure 9 shows the sorted saliencies of a point cloud sample with 33,638 points. As shown in Fig. 9, the extracted saliencies present a piecewise curve pattern, thereby indicating different local geometries. Thus, we use a piecewise-linear spline [25] with four free knots for the segmentation of the surface and curve saliencies and a spline with five free knots for the junction saliency segmentation based on the observations of Fig. 9. The segmentation results and the total number of points within this point cloud sample are shown in Fig. 9. The vertical dash lines and their associated numbers show the segmentation point indexes, and different segments are in different colors. Figure 10 shows the associated surface segmentation, and the color of the segmentation is

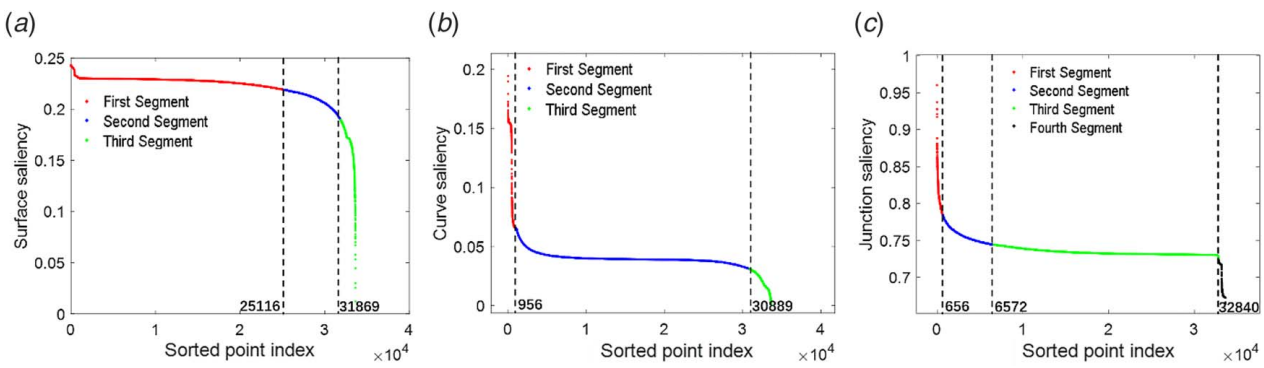

Fig. 9 (*a*) Surface, (*b*) curve, and (c) junction saliencies of one sample (Color version online.)

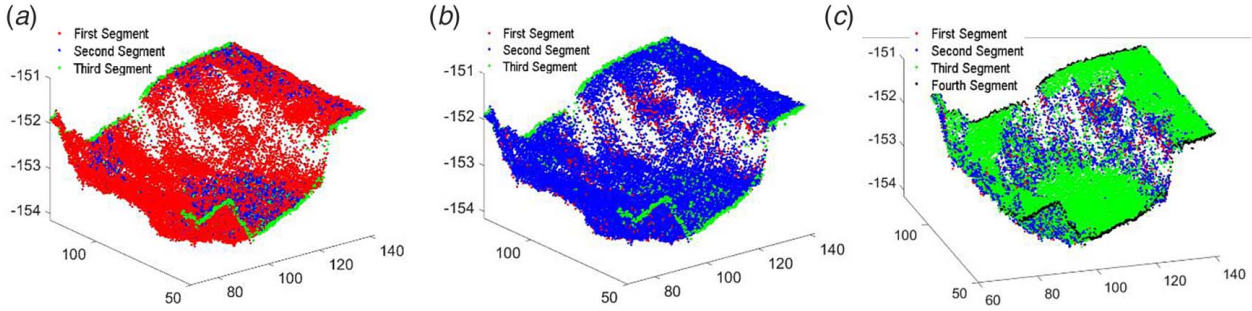

Fig. 10 Surface segmentation of one sample associated with the segmentation of (*a*) surface, (*b*) curve, and (c) junction saliencies (Color version online.)

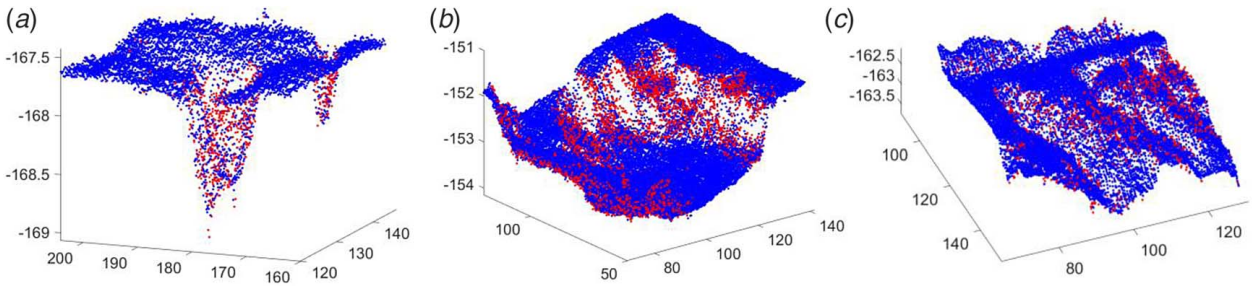

Fig. 11 Sharp points and different types of surface anomalies such as (*a*) a pinhole, (*b*) a depression, and (c) oscillation marks

consistent with that in Fig. 9. As shown in Fig. 10, these three types of geometry saliencies indicate similar local structure information on the steel surface. For example, the first segment of the surface saliency indicates the regular surface points, i.e., the points in the third segment of the junction saliency. Besides, we find the points in the first segment of the junction saliency indicate the sharp profile where surface anomalies may exist. Here, we regard the points in the first segment of the junction saliency as the *sharp points*. Finally, we also select the overlapping points between the second segments of the junction and surface saliencies as sharp points to provide more local structure information. Figure 11 shows the selected sharp points (marked in red) in different types of surface anomalies, where sharp points are often located along the edges of the anomalies.

5.3 Multiclass Surface Anomaly Classification. Given the sharp point set H_k and the regular point set R_k, the features can be extracted for anomaly classification via a multiclass sparse SVM classifier. Fivefold cross validation is used to select λ_0 in the multiclass SVM classifier. In this case study, we set the six quantiles as [0.025, 0.25, 0.50, 0.75, 0.975, 1] to capture one distance distribution, i.e., $[p_8, p_9, p_{10}, p_{11}, p_{12}, p_{13}] = [p_{14}, p_{15}, p_{16}, p_{17}, p_{18}, p_{19}] = [p_{20}, p_{21}, p_{22}, p_{23}, p_{24}, p_{25}] = [0.025, 0.25, 0.50, 0.75, 0.975, 1]$. We will use 80% of the samples as the training data and the remaining 20% as the testing data. To ensure the robustness of the proposed method, we will conduct 100 replications with different splits of the training and testing data from the original mill dataset. For each replication, the training and testing datasets are split in the way of approximately the same proportion of original samples of each class by stratified sampling. Thus, each random split has approximately the same number of training and testing samples for different classes, thereby indicating all the replications have approximately the same number of training and testing samples for each class.

We randomly select ten replications and show the confusion matrix of classification results in Fig. 12. The *x*-axis shows the predicted class label, and the *y*-axis shows the true class label. The number in the blue box along the diagonal is the accurate percentage of correct predictions, while the number in the red box shows the misclassifications. As shown in Fig. 12, the testing results are highly dependent on the split testing samples. For example, all the testing samples can be accurately classified in some splits for Class 2 and Class 3, while some testing samples can be misclassified in some other splits. Thus, we use 100 random splits to show the overall classification accuracy, where the evaluation index is defined as

$$\text{Accuracy} = \frac{1}{n}\sum_{l=1}^{n} I(\widehat{z}_l = z_l) \tag{12}$$

where z_l is the given label and $\widehat{z}_l$ is the predicted label on the testing dataset. To calculate the classification accuracy of the anomaly class i, the accuracy can be similarly defined as

$$\text{Accuracy}_i = \frac{1}{n_i}\sum_{l=1}^{n_i} I(\widehat{z}_l = i),\ i = 1,2,\ldots,5 \tag{13}$$

where $\widehat{z}_l$ is the predicted label on the testing dataset with the given label i. n_i is the number of the testing samples with the given label i.

The mean accuracy, median accuracy, and standard deviation of the classification accuracy through 100 replications are shown in Table 4. The first five rows show the classification results of five different types of surface anomalies, and the last row shows the classification results of the overall data.

As shown in Table 4, the mean or median classification accuracies of Class 2 and Class 3 are lower than those of the other classes. This is because the sample sizes of Class 2 and Class 3, both 18, are much less than those of Class 4 and Class 5. Such a small sample size will also influence the standard deviation of the classification accuracy. As shown in the last column of Table 4, the standard deviation decreases if the sample size increases. However, the overall performance of the proposed approach is satisfactory, with 0.8627 mean accuracy and 0.8667 median accuracy.

Notably, although Class 2 and Class 3 have the same sample size, Class 2 has a lower mean and median accuracy than Class 3. This is because the oscillation mark type of anomaly is more like other

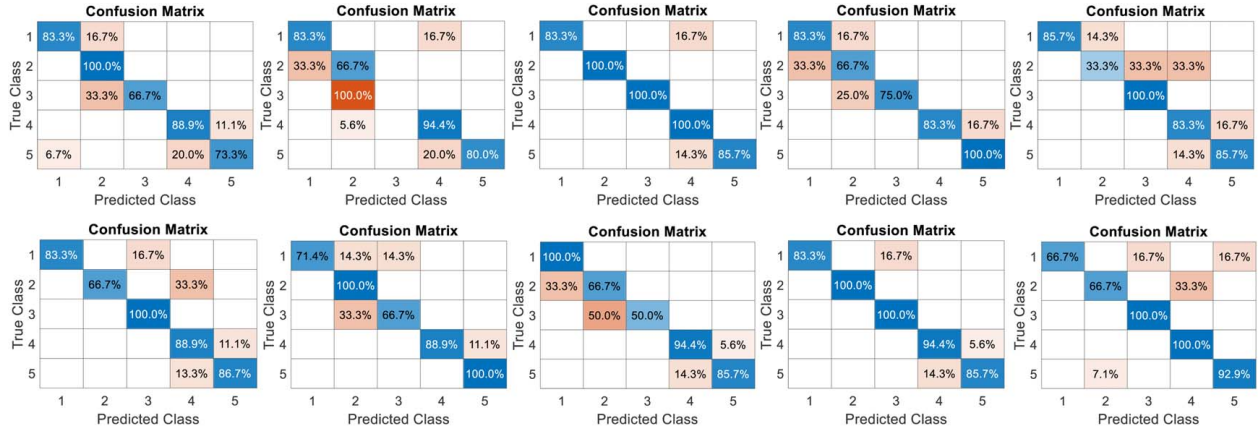

Fig. 12 Confusion matrix of classification results for ten random splits

Table 4 Classification results of 100 random splits on the testing dataset

Anomaly class	Anomaly type	Sample size	Mean accuracy	Median accuracy	Standard deviation
1	Debris patch	30	0.8350	0.8333	0.1629
2	Oscillation mark	18	0.6283	0.6667	0.2849
3	Depression	18	0.7983	0.8750	0.2361
4	Slag	92	0.9154	0.9444	0.0617
5	Pinhole	70	0.8755	0.8667	0.0845
Overall	–	228	0.8627	0.8667	0.0507

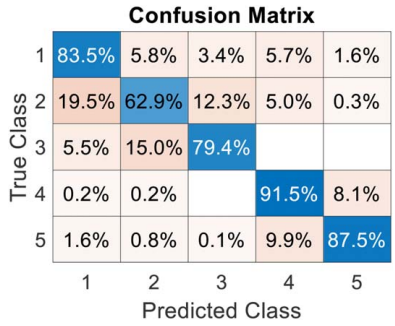

Fig. 13 The confusion matrix of classification results for the summation of 100 random splits

anomaly patterns, such as depression and debris patch. Figure 13 shows the confusion matrix that presents the accurate and misclassified percentage of predictions on the testing dataset from overall 100 splits. As shown in Fig. 13, the oscillation mark type of anomalies can be misclassified as debris patch or depression in several cases. In comparison, the depression type of anomaly is less likely to be misclassified as other types of anomalies (especially the pinhole and slag).

To further check the effective features that are selected for classification, we discuss the features that have nonzero coefficients. First, if one feature is not important, then all the related coefficients of K-class functions will be zero, implemented by the sup-norm penalty [31]. Thus, the selection frequencies count only except the feature that all the related coefficients of K-class functions are zero. The results of selection frequencies on these 100 random splits are reported in Table 5.

As shown in Table 5, features $f_2 \sim f_7$ are usually selected for most random splits, which are the standard deviations of location coordinates of sharp points and regular points. These features provide more information, so the percentage of sharp points, i.e., feature f_1, is not commonly selected due to less additional information. In addition, $f_{12}, f_{13}, f_{19}, f_{25}$ are selected in each split, which are the large quantiles of distance distribution from the artifact surface in 3D for regular surface points and anomaly points. This is because the large quantile of distance distribution can show the depth of the anomaly for the anomaly points and the large noise for the regular surface. The middle quantiles are also sometimes selected to represent more information about the regular surface points and the sharp anomaly points. In comparison, the small quantile features f_8, f_{14}, f_{20} are never selected in the 100 random splits.

To further check the effective features of each anomaly class, we also list the frequency of nonzero coefficients of the K-class functions in Table 6. We also show the frequency of selected features for different classes and overall classification in Fig. 14. As shown in Table 6 and Fig. 14, most classes use features $f_2 \sim f_7, f_{12}, f_{13}, f_{19}, f_{25}$, to identify the related anomaly. But we can find that Class 1 (debris patch) less uses f_4, i.e., the standard deviation of the z coordinate, compared with other classes. In addition, Class 2 (oscillation mark) less uses f_{16}, i.e., the middle quantile of distances of sharp points to distinguish the oscillation mark anomaly. However, these features are overall very important to identify the different types of anomalies.

5.4 Sensitivity Analysis. In this section, sensitivity analysis is performed in terms of the neighborhood size M in tensor voting. We set M as 100, 200, 300, 400, 500, respectively, to test the performance of the overall approach. The 100 random splits and other

Table 5 Selection frequencies of each feature over 100 random splits

Feature	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}
Frequency	8	100	100	95	100	100	95	0	2	5	85	100	100
Feature	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}	
Frequency	0	3	52	88	75	100	0	4	4	46	91	100	

Table 6 Nonzero coefficient frequencies of each feature over 100 random splits

Feature	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}
Class 1	2	100	100	60	100	100	68	0	2	5	85	100	99
Class 2	7	100	100	95	100	100	77	0	2	5	72	87	98
Class 3	8	100	100	93	100	99	94	0	2	1	71	100	97
Class 4	8	100	100	94	100	100	95	0	2	5	77	100	100
Class 5	8	100	100	95	100	100	94	0	2	5	82	100	100

Feature	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}
Class 1	0	3	52	88	72	99	0	4	4	36	45	100
Class 2	0	3	7	81	69	99	0	3	3	41	83	99
Class 3	0	0	51	59	46	96	0	3	3	46	90	100
Class 4	0	3	52	84	71	100	0	4	4	44	89	100
Class 5	0	3	51	87	66	99	0	4	4	36	91	100

parameter settings are the same as above. Figure 15 shows results of sensitivity analysis results from Class 1 to Class 5 and overall accuracy results. The detailed results are listed in Table 7. As shown in Fig. 15, the mean and median classification accuracy of Class 3 has been improved for the neighborhood size $M = 400$. However, such improvement is very limited in terms of overall accuracy. Notably, tensor voting is mainly used to extract the local geometry information for sharp point selection, and the features are extracted globally given the sharp points and regular points. Thus, the classification results are not sensitive to the neighborhood size.

5.5 Comparison Studies. To further show the effectiveness of our extracted features and the proposed approach overall, we compare our features with the histogram features from Ref. [14]. The deviations between point cloud data and the nominal model are used for feature extraction and further classification in Ref. [14]. In this paper, we do not have a typical nominal model for forming process, so we assume the nominal model is the fitted plane to make comparisons. We mainly focus on the comparison between our extracted features and the features in Ref. [14], so we use the same type of sparse multiclass SVM classifier for comparison to demonstrate the effectiveness of tensor voting-based feature extractions and the accuracy of the proposed approach overall.

The histogram features of deviation distance from point cloud data to the nominal model, such as the mean, mode, and width, are used as features in Ref. [14]. In addition, the Mahalanobis distance values between the computed histogram and baseline histogram are also used as features in Ref. [14]. Thus, there are four types of features used in Ref. [14] to classify the anomalies. The baseline histogram is assumed with $N(0, \sigma^2)$ distribution, where the variance is set to be an ideal variation. We set the parameter $\sigma = 0.002$, which is the same as Ref. [14]. In addition, we do not have the observed random deviation or systematic deviation, so we only use the overall distance deviations of points to establish the observation histogram to calculate the distance as the feature. We also use 20 bins to establish the histogram, which is the same as Ref. [14]. Notably, the covariance matrix between different bins is meaningless if we use all the anomaly datasets to estimate, so we use the Euclidean distance values as the features. We also consider the frequency of the deviation per bin and bin-edge values as the variables that define histograms in Ref. [14]. We

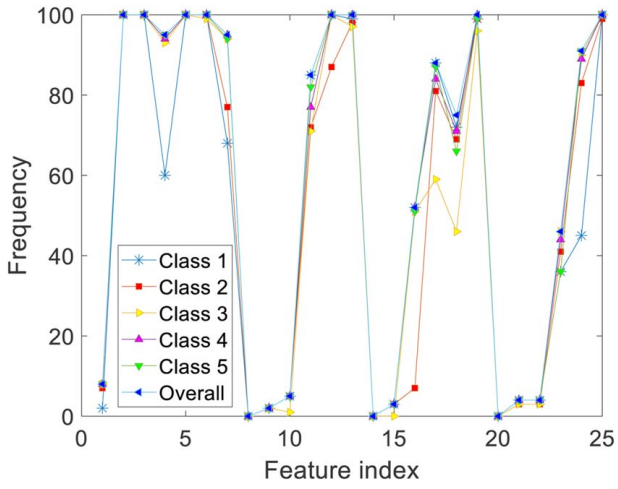

Fig. 14 Illustration of frequencies of selected features in different classes and overall classification

Table 7 Sensitivity analysis results of 100 random splits on the testing dataset

Anomaly class	Anomaly type	Neighborhood size	Mean accuracy	Median accuracy	Standard deviation
1	Debris patch	100	0.8333	0.8333	0.1704
		200	0.8207	0.8333	0.1763
		300	0.8350	0.8333	0.1629
		400	0.8336	0.8333	0.1650
		500	0.8290	0.8333	0.1715
2	Oscillation mark	100	0.6425	0.6667	0.3044
		200	0.6158	0.6667	0.2879
		300	0.6283	0.6667	0.2849
		400	0.6225	0.6667	0.2967
		500	0.6150	0.6667	0.2987
3	Depression	100	0.8042	0.8750	0.2220
		200	0.8017	0.8750	0.2332
		300	0.7983	0.8750	0.2361
		400	0.8192	1.0000	0.2238
		500	0.8092	1.0000	0.2328
4	Slag	100	0.9131	0.9444	0.0626
		200	0.9153	0.9444	0.0594
		300	0.9154	0.9444	0.0617
		400	0.9137	0.9444	0.0608
		500	0.9153	0.9444	0.0582
5	Pinhole	100	0.8776	0.8667	0.0800
		200	0.8817	0.8976	0.0764
		300	0.8755	0.8667	0.0845
		400	0.8739	0.8667	0.0826
		500	0.8797	0.9286	0.0750
Overall	–	100	0.8636	0.8667	0.0492
		200	0.8618	0.8667	0.0493
		300	0.8627	0.8667	0.0507
		400	0.8624	0.8667	0.0501
		500	0.8629	0.8667	0.0476

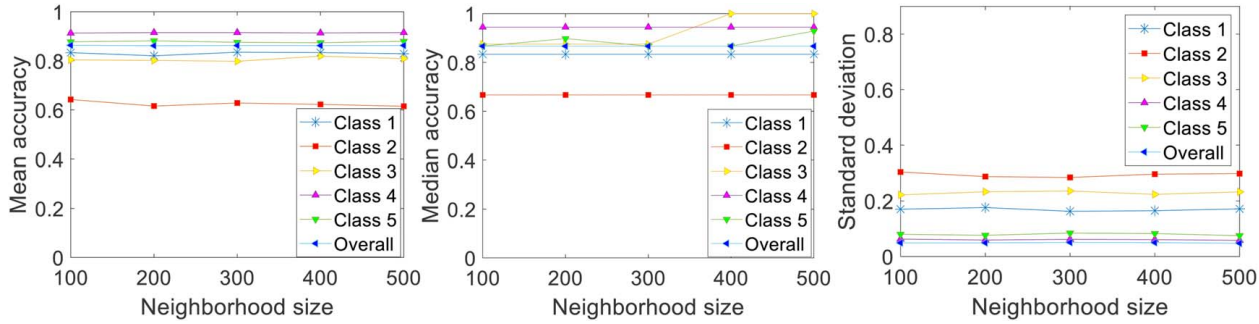

Fig. 15 Classification results of the testing dataset from 100 random splits given different neighborhood sizes

Table 8 Comparison results of 100 random splits on the testing dataset

Anomaly class	Anomaly type	Sample size	Method	Mean accuracy	Median accuracy	Standard deviation
1	Debris patch	30	**Ours**	**0.8350**	**0.8333**	**0.1629**
			Histogram	0.7514	0.8333	0.1819
2	Oscillation mark	18	**Ours**	**0.6283**	**0.6667**	**0.2849**
			Histogram	0.4867	0.5000	0.2941
3	Depression	18	**Ours**	**0.7983**	**0.8750**	**0.2361**
			Histogram	0.1508	0.0000	0.2071
4	Slag	92	**Ours**	**0.9154**	**0.9444**	**0.0617**
			Histogram	0.8489	0.8333	0.0803
5	Pinhole	70	**Ours**	**0.8755**	**0.8667**	**0.0845**
			Histogram	0.8462	0.8571	0.0939
Overall	–	228	**Ours**	**0.8627**	**0.8667**	**0.0507**
			Histogram	0.7589	0.7556	0.0471

Note: Bold signifies our results compared to the benchmark methods.

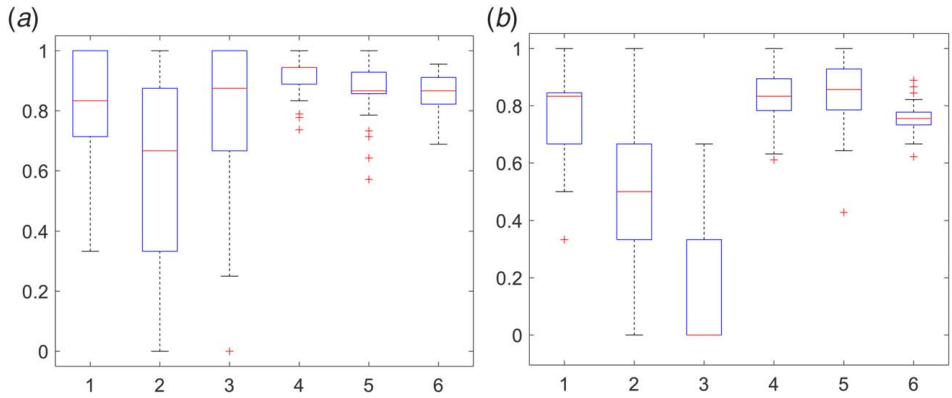

Fig. 16 Boxplots of classification accuracy on the testing dataset from 100 random splits: (*a*) our method and (*b*) histogram method [14]

use the same 100 splits of training and testing dataset, and the same type of sparse multiclass SVM classifier for a fair comparison.

The classification results are listed in Table 8, where the method of extracted histogram features is denoted as "histogram" and our method of tensor voting-based features is denoted as "ours." As shown in Table 8, our tensor voting-based features are more effective than histogram features, resulting in a significant improvement in the classification accuracy. To illustrate the classification results more comprehensively, Fig. 16 shows the boxplots of the classification accuracies for all five classes of anomalies in Table 8. Our features outperform histogram features in all types of anomalies in terms of mean accuracy and have a smaller standard deviation except for depression type of anomalies. However, histogram features perform badly on depression anomalies, which leads to 0 median accuracy. Therefore, our tensor voting-based feature extractions are more effective overall.

5.6 Discussion. Several limitations can be further improved in future work. Currently, we assume the surface is a plane, and use SVD to fit the plane. When the point size is large, this step will be very computationally intensive. Downsampling techniques can be adopted for plane fit. In the above case study, when the regular point size is over 40,000, half the number of points are uniformly sampled along the x-axis for plane fit. How to relax the plane assumption of surface to make the proposed methodology more applicable to a free-form surface will be a potential direction for future work. Another limitation of this work is that only one type of anomaly is predicted. In practice, one surface may have several types of anomalies, but only one label is given per surface in the training dataset. How to predict multiple anomalies on one surface under a noisy label is also worth investigations. In addition, the approach documented in this paper can predict the anomaly types carried on the surface of a manufactured artifact. Both the

detection and classification of multiple types of surface anomalies are also worth investigation.

6 Conclusion

A novel tensor voting-based classification approach for surface anomaly classification using 3D point cloud data is proposed. This approach directly extracts useful features by way of saliency derivation, sharp point selection, and aggregate data statistics from the massive 3D unstructured point cloud data without any transforming or dimensional reduction in data. The features can be used in a sparse multiclass SVM classifier for surface anomaly classification. A case study illustrates the effectiveness of the proposed approach, despite some limits due to the sample sizes of two subsets. Comparison studies are also performed to further demonstrate the effectiveness of the proposed approach.

Acknowledgment

The authors would like to thank the five anonymous reviewers for their constructive comments and valuable suggestions, which significantly improves the earlier version of this paper.

Conflict of Interest

There are no conflicts of interest.

References

[1] Popa, E. M., and Kiss, I., 2011, "Assessment of Surface Defects in the Continuously Cast Steel," Acta Tech. Corviniensis-Bull. Eng., **4**(4), p. 109.

[2] Bao, L., Wang, K., and Jin, R., 2014, "A Hierarchical Model for Characterising Spatial Wafer Variations," Int. J. Prod. Res., **52**(6), pp. 1827–1842.

[3] Colosimo, B. M., Cicorella, P., Pacella, M., and Blaco, M., 2014, "From Profile to Surface Monitoring: SPC for Cylindrical Surfaces via Gaussian Processes," J. Qual. Technol., **46**(2), pp. 95–113.

[4] Dastoorian, R., Elhabashy, A. E., Tian, W., Wells, L. J., and Camelio, J. A., 2018 "Automated Surface Inspection Using 3D Point Cloud Data in Manufacturing: A Case Study," ASME 2018 13th International Manufacturing Science and Engineering Conference, College Station, TX, June 18–22, p. V003T02A036.

[5] Yan, H., Pacella, M., and Paynabar, K., 2019, "Structured Point Cloud Data Modeling via Regularized Tensor Decomposition and Regression," Technometrics, **61**(3), pp. 385–395.

[6] Gahrooei, M. R., Yan, H., Paynabar, K., and Shi, J., 2020, "Multiple Tensor on Tensor Regression: An Approach for Modeling Processes With Heterogeneous Sources of Data," Technometrics, **63**(2), pp. 147–159.

[7] Wells, L. J., Megahed, F. M., Niziolek, C. B., Camelio, J. A., and Woodall, W. H., 2013, "Statistical Process Monitoring Approach for High-Density Point Clouds," J. Intell. Manuf., **24**(6), pp. 1267–1279.

[8] Colosimo, B. M., Mammarella, F., and Petro, S., 2010, "Quality Control of Manufactured Surfaces," *Frontiers in Statistical Quality Control 9*, H. J. Lenz, P. T. Wilrich, and W. Schmid, eds., Springer, New York, pp. 55–70.

[9] Huang, D., Du, S., Li, G., Zhao, C., and Deng, Y., 2018, "Detection and Monitoring of Defects on Three-Dimensional Curved Surfaces Based on High-Density Point Cloud Data," Precis. Eng., **53**, pp. 79–95.

[10] Ren, J., and Wang, H., 2019, "Surface Variation Modeling by Fusing Multiresolution Spatially Nonstationary Data Under a Transfer Learning Framework," ASME J. Manuf. Sci. Eng., **141**(1), p. 011002.

[11] Wells, L. J., Shafae, M. S., and Camelio, J. A., 2016, "Automated Surface Defect Detection Using High-Density Data," ASME J. Manuf. Sci. Eng., **138**(7), p. 071001.

[12] Zhao, X., and Del Castillo, E., 2020, "An Intrinsic Geometrical Approach for Statistical Process Control of Surface and Manifold Data," Technometrics, **63**, pp. 1–18.

[13] Samie Tootooni, M., Dsouza, A., Donovan, R., Rao, P. K., Kong, Z. J., and Borgesen, P., 2017, "Classifying the Dimensional Variation in Additive Manufactured Parts From Laser-Scanned Three-Dimensional Point Cloud Data Using Machine Learning Approaches," ASME J. Manuf. Sci. Eng., **139**(9), p. 091005.

[14] Yacob, F., Semere, D., and Nordgren, E., 2019, "Anomaly Detection in Skin Model Shapes Using Machine Learning Classifiers," Int. J. Adv. Manuf. Technol., **105**, pp. 1–13.

[15] Himmelsbach, M., Luettel, T., and Wuensche, H.-J., 2009, "Real-Time Object Classification in 3D Point Clouds Using Point Feature Histograms," 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp. 994–1000.

[16] Wohlkinger, W., and Vincze, M., 2011, "Ensemble of Shape Functions for 3D Object Classification," 2011 IEEE International Conference on Robotics and Biomimetics, Karon Beach, Thailand, Dec. 7–11, pp. 2987–2992.

[17] Qi, C. R., Su, H., Mo, K., and Guibas, L. J., 2017, "Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, July 21–26, pp. 652-660.

[18] Rodríguez-Cuenca, B., García-Cortés, S., Ordóñez, C., and Alonso, M., 2015, "Automatic Detection and Classification of Pole-Like Objects in Urban Point Cloud Data Using an Anomaly Detection Algorithm," Remote Sens., **7**(10), pp. 12680–12703.

[19] Qi, C. R., Yi, L., Su, H., and Guibas, L. J., 2017, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," The Neural Information Processing Systems, Long Beach, CA, June 7, pp. 5105–5114.

[20] Zhao, H., Jiang, L., Fu, C.-W., and Jia, J., 2019, "PointWeb: Enhancing Local Neighborhood Features for Point Cloud Processing," The Computer Vision and Pattern Recognition, Long Beach, CA, June 15, pp. 5560–5568.

[21] Guy, G., and Medioni, G., 1997, "Inference of Surfaces, 3D Curves, and Junctions From Sparse, Noisy, 3D Data," IEEE Trans. Pattern Anal. Mach. Intell., **19**(11), pp. 1265–1277.

[22] Medioni, G., Tang, C.-K., and Lee, M.-S., 2000, "Tensor Voting: Theory and Applications," Proceedings of RFIA, Paris, France, vol. 3.

[23] Wu, T. P., Yeung, S.-K., Jia, J., Tang, C.-K., and Medioni, G., 2012, "A Closed-Form Solution to Tensor Voting: Theory and Applications," IEEE Trans. Pattern Anal. Mach. Intell., **34**(8), pp. 1482–1495.

[24] Granlund, G. H., and Knutsson, H., 1995, *Signal Processing for Computer Vision*, Springer, Berlin.

[25] D'Errico, J., "Shape Language Modeling." https://www.mathworks.com/matlabcentral/fileexchange/24443-slm-shape-language-modeling, Accessed April 27, 2018.

[26] Camisani-Calzolari, F., Craig, I., and Pistorius, P., 2003, "A Review on Causes of Surface Defects in Continuous Casting," IFAC Proc. Vol., **36**(24), pp. 113–121.

[27] Söderkvist, I., 2009, "Using SVD for Some Fitting Problems," University Lecture, https://www.ltu.se/cms_fs/1.51590!/svd-fitting.pdf, Accessed April 27, 2018.

[28] Boser, B. E., 1992, "A Training Algorithm for Optimal Margin Classifiers," Proceedings of the Fifth Annual Workshop on Computational Learning Theory, Pittsburgh, PA, July 27–29, ACM, Vol. 5, pp. 144–152.

[29] Cortes, C., and Vapnik, V., 1995, "Support-Vector Networks," Mach. Learn., **20**(3), pp. 273–297.

[30] Lee, Y., Lin, Y., and Wahba, G., 2004, "Multicategory Support Vector Machines: Theory and Application to the Classification of Microarray Data and Satellite Radiance Data," J. Am. Stat. Assoc., **99**(465), pp. 67–81.

[31] Huang, L., Hao, H. Z., Zeng, Z. B., and Bushel, P. R., 2013, "Improved Sparse Multi-Class SVM and Its Application for Gene Selection in Cancer Classification," Cancer Inf., **12**(12), pp. 143–153.

[32] Crammer, K., and Singer, Y., 2002, "On the Algorithmic Implementation of Multiclass Kernel-Based Vector Machines," J. Mach. Learn. Res., **2**(2), pp. 265–292.

[33] Wu, T. P., Yeung, S. K., Jia, J., and Tang, C. K., 2010, "Quasi-Dense 3D Reconstruction Using Tensor-Based Multiview Stereo," 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, June 13–18, pp. 1482–1489.