ANTLER: Bayesian Nonlinear Tensor Learning and Modeler for Unstructured, Varying-Size Point Cloud Data

Michael Biehler, Member, IEEE, Hao Yan, and Jianjun Shi

Abstract—Unstructured point clouds of varying sizes are increasingly acquired in a variety of environments through laser triangulation or Light Detection and Ranging (LiDAR). Predicting a vector response based on unstructured point clouds is a common problem that arises in a wide variety of applications. The current literature relies on several pre-processing steps such as structured subsampling and feature extraction to analyze the point cloud data. Those techniques lead to quantization artifacts and do not consider the relationship between the regression response and the point cloud during pre-processing. Therefore, we propose a general and holistic “Bayesian Nonlinear Tensor Learning and Modeler” (ANTLER) to model the relationship of unstructured, varying-size point cloud data with a vector response. The proposed ANTLER simultaneously optimizes a nonlinear tensor dimensionality reduction and a nonlinear regression model with a 3D point cloud input and a regression response. ANTLER can consider the complex data representation, high-dimensionality, and inconsistent size of the 3D point cloud data.

Note to Practitioners—This paper is motivated by a real-world case study concerning the prediction of the transmission error and eccentricity based on unstructured point clouds of varying sizes in gear manufacturing. In the current state-of-the-art method, those characteristics can only be obtained via expensive and time-consuming Finite Element Analysis (FEA) or test benches. The proposed ANTLER framework can directly link the measurement point clouds with a vector response and serves as a guiding example for the immense potential of the ANTLER.

Index Terms—Unstructured point cloud data, nonlinear point cloud regression, nonlinear tensor decomposition, high-dimensionality modeling.

I. INTRODUCTION

THREE-DIMENSIONAL (3D) point cloud acquisition devices, such as laser scanners for surface modeling and geometric reconstruction, have created vast amounts of unstructured 3D point clouds. Previous work has mainly addressed the segmentation [1] and classification [2] of objects based on point clouds. However, the nonlinear modeling of high-dimensional point cloud data in the engineering domain with limited sample sizes has received little attention. To this end, it is necessary to propose a holistic one-step framework that can combine extraction of nonlinear features and estimation of a nonlinear relationship between unstructured point clouds and a vector response. Such modeling problems based on point clouds have essential applications, for example, in the manufacturing domain. In additive manufacturing (AM), the prediction of part distortion [3] based on the 3D layer shapes for a given process setting relies heavily on time-consuming FEA, which is infeasible for in-process functional quantification and verification. A holistic machine learning approach to directly predict the response (e.g., part distribution) is needed for in-process monitoring. The application that motivated this research is the prediction of the transmission error of gears based on in-line 3D measurement data (Fig. 1). An essential key quality indicator for gears to quantify the manufacturing shape inaccuracies is the transmission error. In the current state-of-the-art, the modeling of the relationship between the shape and the transmission error depends on expensive experiments or time-consuming FEA simulations. The shape inaccuracies and deviations are the main determining factor for key quality indicators such as the transmission error. Therefore, the shape of each manufactured gear is measured with optical metrology, resulting in a 3D point cloud. Then we directly relate those 3D shapes represented by point clouds to the resulting transmission error via our ANTLER framework.

Fig. 1. Illustration of the unstructured point cloud from in-line 3D measurement and transmission error.
To calculate the transmission error for one manufactured gear, the gear is meshed with a “master” gear, representing the ideal gear design. Therefore, the measured transmission error will be solely caused by the manufactured gear and not by the master gear it is meshed with. For details on this procedure, we refer interested readers to Vincent et al. [4].

The point clouds presented in the micro gear case study (Section V) exhibit a highly unstructured point distribution (see zooms in Figure 1) due to the acquisition by optical metrology. They are not organized on a structured grid but exhibit an irregular structure, with no clear pattern. The acquisition technique also results in a varying number of measurement points across samples.

The main goal of this article is to propose a novel nonlinear regression method to quantify the relationship between unstructured, varying-size point clouds and a scalar or multivariate, vector response. However, there are significant challenges in achieving this objective due to the complex data format of the unstructured 3D point clouds:

- **Unstructured**: Typically, there is no prior structure of the set of 3D coordinates for complex shapes. In particular, their data structure is not topologically aware, and the spatial neighboring relationship among points is unknown, which leads to significant modeling challenges for nonlinear point cloud regression. For unstructured point cloud data, each point is independent and the distance to neighboring points is not fixed as illustrated in Fig. 2 a). Furthermore, the distribution of points is irregular: widely used devices such as laser scanners or LiDAR lead to sparse and dense point cloud regions (Fig. 2 b) based on the acquisition conditions (e.g., illumination).

- **High dimensionality**: Point clouds are spatially dense, containing a large number of measurement points (e.g., millions) in each sample, which poses significant computational challenges.

- **Limited sample size**: On the contrary, the number of samples in engineering applications is typically very limited and much smaller than the number of dimensions.

- **Varying size**: The number of measurement points varies per sample, making the adaptation of machine learning techniques challenging because most machine learning models require designed input training data with a regular structure and fixed dimensionality. The reason for the varying number of measurement points lies in the acquisition by commonly used devices such as laser scanners or LiDAR. Those devices will detect a different number of points based on the measurement conditions (e.g., illumination, rain). For a detailed discussion, we refer interested readers to Fukuchi and Shiina [5] and Filgueira et al. [6].

To tackle those challenges, existing research has mainly focused on combining dimensionality reduction and regression models for structured point cloud data. The structure allows the efficient representation of the point clouds as tensors [7]. However, these methods cannot be applied directly to unstructured point cloud data without subsampling or interpolation, which typically only preserves global spatial information while losing local spatial information. In particular, detail-level spatial information such as local features or anomalies are lost. For a more detailed review of the tensor-based structured point cloud methods, readers are referred to Section II. A.

Recently, several deep learning-based methods have been proposed for unstructured point cloud data [1], [8]. These methods can learn the low-dimensional representation directly from the raw point cloud data but require a large number of training samples, which may not be feasible in engineering applications. For a more detailed review of the deep-learning-based point cloud data regression and classification, please refer to Section II. C.

To address these aforementioned challenges, we propose a novel method, namely “Bayesian Nonlinear Tensor Learning and Modeler” (ANTLER), to model the unstructured, varying-size point cloud data and a scalar or multivariate, vector response is introduced in Section III. Section IV validates the proposed ANTLER framework for an unstructured, varying-size 3D point cloud input and a vector response is introduced in Section III. Section IV validates the proposed methodology by using simulated data with two different types of unstructured point clouds. Furthermore,
the performance of the proposed method is compared with existing benchmark methods in terms of estimation accuracy and computational time. In Section V, we conduct a real-world case study for predicting the functional response in gear manufacturing applications. Finally, we conclude the article with a short discussion and an outline of future work in Section VI.

II. LITERATURE REVIEW

In this section, we will review three major categories of methods for point cloud modeling problems, including tensor regression, feature extraction-based, and deep learning methods. We note, to the best of our knowledge, that no existing research has directly addressed regression problems with unstructured, varying-size 3D point clouds as inputs and scalar or multivariate responses. A wide variety of machine learning methods, such as traditional regression methods, do not apply to this problem since vectorized unstructured point clouds are not permutation invariant. Therefore, their spatial relationship cannot be preserved because the order of measured locations in the vectorized format may change with every sample. Consequently, the remainder of this section only reviews methods that can be extended to model the unstructured point clouds. Most of the latter methods cannot implicitly handle input point clouds of varying sizes.

A. Tensor Regression Techniques for Structured Point Cloud Data

While classical regression techniques treat covariates as vectors, the rise of sensing technology and high-dimensional (heterogeneous) data has led to covariates of more complex forms such as multidimensional arrays, also known as tensors. However, these methods require the efficient representation of the data as a tensor. Therefore, in the domain of point cloud data, the data type of structured point cloud has been the focus of attention. In structured point clouds, the data lies in a predefined grid and can be efficiently represented as a tensor. In this direction, Yan et al. [7] proposed a regularized tensor regression for structured point cloud data analysis, which links variational patterns of point clouds (response) and to the process variables (inputs). Gahrooei et al. [10] have proposed a multiple tensor-on-tensor (MTOT) regression framework to model the behavior of a system as a function of heterogeneous sets of data, such as scalars, waveform signals, images, or structured point clouds. Wang et al. [11] proposed an augmented tensor regression model for settings with missing data. By integrating tensor regression and tensor completion, they can handle scenarios where measurements of the 3D point clouds are incomplete.

In the field of nonlinear tensor regressions, Kossaifi et al. [12] developed a novel neural network component, namely Tensor Regression Layers (TRL), which map high-order tensors to an output of arbitrary order. TRL leverages the multilinear structure of the input tensor by enforcing a low multilinear rank of the regression weight tensor. This allows the projection to a low-rank subspace that jointly models the input and the output. In this direction, Kim et al. [13] proposed to apply Tucker decomposition to the convolutional kernel tensors of a pretrained network. Novikov et al. [14] compress fully connected layers by imposing a low-rank tensor structure via the Tensor-Train format. Imaizumi and Hazashi [15] proposed a nonparametric tensor regression that models the nonlinearity by decomposing the regression function into simple local functions by incorporating low-rank tensor decomposition. Zhou et al. [16] utilize polynomial splines and elastic net penalization to model the nonlinearity in tensor regression nonparametrically.

However, these methods have the limitation that they are generic tensor regression approaches, which do not consider the specific spatial structure and characteristics of 3D point clouds.

B. Feature Extraction-Based Approach for Point Cloud Data

The current literature on point cloud modeling mainly utilizes a two-step approach, which first extracts useful features and then uses them as input to an off-the-shelf predictive model. A wide range of features such as dissimilarity metrics [17], principal components [18], or geometric features [19] have been proposed to this end. Furthermore, several methods are based on tensor voting [2], [20], [21]. For example, Du et al. [2] proposed a tensor voting-based surface anomaly classification approach based on 3D point cloud data. Features obtained from the point clouds via saliency derivation and sharp point selection are used to train a sparse multiclass surface anomaly classifier.

C. Deep Learning Methods for Point Cloud Data

Recently, deep learning methods have been developed for classification and object segmentation of the point cloud data due to a large amount of public 3D point cloud datasets. Most prominently, [1] proposed a deep neural network (PointNet) that uses point cloud inputs for various 3D recognition tasks such as object classification, part segmentation, and semantic segmentation. Early works utilize the distribution of the points to aid the classification of 3D objects [22]. Another direction of deep learning research utilizes graph neural networks for object detection [8], classification [23], and segmentation [24]. In particular, [8] encodes the point cloud efficiently in a fixed radius near-neighbors graph and then utilizes a graph neural network to predict the category and shape of objects. Similarly, [23] proposed a graph attention-based point neural network to learn shape representations from point clouds. [24] utilized spectral graph theory to represent point cloud features as signals on a graph and applied a graph convolutional neural network on the graph Laplacian matrix.

Recently, a wide range of new deep learning methods have been proposed to classify or segment the point cloud data. PointNet++ [25] was introduced as an extension to the seminal PointNet paper. Grid-GCN [26] uses a novel data structuring strategy to improve spatial coverage while reducing the time complexity for fast and scalable point cloud learning.

RepSurf-U [27] is the current state-of-the-art benchmark on ModelNet40. RepSurf-U uses a novel representation, that explicitly learns local structures in point clouds, which is
highly relevant for regression tasks. Here, we refer readers to Guo et al. [9] for a detailed review of deep learning in 3D point cloud learning.

The major drawback of those deep learning methods is the difficulty in capturing sparse local features and that they are not sample efficient. Therefore, they are not suitable for engineering applications with small engineering tolerance and consequent high precision requirements for discriminative models. Additionally, those techniques can lead to computational challenges due to the high dimensionality of the point clouds. The large number of model parameters to be estimated can also require a very large number of training samples. Therefore, it may lead to overfitting or inaccurate models, especially when the number of training samples is small.

In summary, there is very limited work on regressions with unstructured point cloud input and vector output, which can address the aforementioned challenges. This paper fills these research gaps and proposes a method for the prediction of scalar or vector responses based on an unstructured, varying-size point cloud input.

III. ANTLER METHODOLOGY

In this section, we introduce the ANTLER framework as an approach to model a vector response as a function of an unstructured, varying-size 3D point cloud input.

We assume that a set of unstructured 3D input point clouds of size $N$ is available, denoted as $\mathbf{X}_i \in \mathbb{R}^{3 \times M_i}$ of varying-size $M_i$ and a response vector $\mathbf{Y}_i \in \mathbb{R}^p$, where $p$ is the number of outputs, and $i$ is the sample index. The ANTLER framework first represents the unstructured point cloud $\mathbf{X}$ as a binary tensor $\mathbf{B}$ utilizing a voxel-based representation (Section III.A). Afterward, an efficient sampling strategy (Section III.B) is utilized to obtain a balanced sampling tensor $\mathbf{D}$.

Subsequently, a Streaming Nonlinear Bayesian Tensor Decomposition (SNBTD [28]) (Section III.C) is utilized to obtain a low dimensional embedding $\mathcal{U}$ of the binary tensor $\mathbf{B}$. In particular, random Fourier features are used to obtain a sparse spectrum Gaussian Process (GP) decomposition model. In Section III.D, we link the low dimensional embedding $\mathcal{U}$ and the vector response $\mathbf{Y}$ via a nonlinear regression method. In particular, $\mathbf{Y} = g_{\theta_r}(\mathcal{U}) + \mathbf{E}$, where $\theta_r$ are the model parameters to be estimated, and $\mathbf{E}$ is the error tensor.

Finally, we utilize a Variational Autoencoder (VAE) with a novel ANTLER loss to simultaneously learn the low dimensional embedding $\mathcal{U}$ and find the nonlinear regression relationship between $\mathcal{U}$ and the response $\mathbf{Y}$. Interested readers are referred to Doersch [29] and the references therein for a detailed review of the VAE framework.

The unified framework for the simultaneous optimization of the decomposition of the binary tensor $\mathbf{B}$ into the low dimensional embedding $\mathcal{U}$ and regression function (i.e., $\mathbf{Y} = g(\mathcal{U})$) enables manifold learning (Section III.E) to capture the essential information necessary for the discriminative task and optimize the proposed representation. This procedure is summarized in Figure 3.

The main underlying assumption for the ANTLER framework is the manifold hypothesis, which states that high-dimensional observations typically lie in a much lower-dimensional, not necessarily linear, manifold.

A. Binary 3D Tensor Representation of Point Cloud Data

Several approaches to represent surfaces of 3D objects, such as triangular meshes or graphs, have been proposed in the literature. However, there are three main requirements regarding the representation of unstructured, varying-size point clouds:

1. Representation should consider the spatial correlation structure in unstructured point clouds.
2. Representations should respect the permutation invariance of the input point clouds [1].
3. Representations should have fixed dimensions instead of the original varying dimensions of the point cloud data.

Therefore, we take advantage of the volumetric voxel representation of point clouds [30]. This approach allows the transformation of unstructured point clouds into regular voxel grids. Each voxel contains a Boolean occupancy status (i.e., occupied by measurement point or unoccupied).

In our setting, the original point cloud $\mathbf{X}_i$ is represented as a binary 3D tensor $\mathbf{B}_i$ utilizing a 3D grid sufficiently larger than the measurement object to ensure all measurement data is contained inside the grid (Fig. 4a). On this cube, a 3D structured grid with a prespecified grid width and height is defined. Then each grid will be assigned binary values: If there are measurement points in $\mathbf{B}_i$, the value in that particular grid region $\mathbf{B}_i$ is assigned as 1. If there is no measurement point in a region, the corresponding region will have entry 0 (Fig.4b and 4c).

The recommended strategy for selecting the grid size is as follows: The grid size is determined by starting from a prespecified (small) grid size (e.g., $100 \times 100 \times 100$). Then, the resolution of the grids is increased until the number of non-zero elements in the binary tensor is equal to the number of measurement points in each training sample. While representing the point cloud in a grid structure is a discretization or approximation technique, our sampling strategy of selecting one measurement point per cube ensures that no significant
information loss occurs. The illustration in Figure 5 shows that with increasing voxel grid resolution the information loss of the voxel representation compared to the original point cloud vanishes.

We note that due to the subsequent balanced sampling (Algorithm 1) the grid size of each sample is not required to be the same for each sample.

The main reasons for selecting the voxel-based representation are two-fold: Firstly, they allow the efficient estimation of occupied and unoccupied space from a wide range of measurements, even when the measurement conditions (e.g., acquisition angle, lighting conditions) are changing. Secondly, they can be stored and manipulated with efficient data structures such as tensors. While this process increases the dimensionality of the data, it keeps the spatial information of the point cloud and is a remedy for the unstructured and non-zero entries. The key idea of the sampling strategy is to represent the empty voxel entries by reconstructing all entries as zero. For example, in the case study presented in Section V, we can achieve an average accuracy of 99.7% by only reconstructing zero entries since the binary only contains 0.3% non-zero entries on average.

**Algorithm 1 Binary Tensor to 4D Array Sampling Strategy**

**Input:**
- Binary Tensor representation \( B_i \) (with entries \( b_j \)) of unstructured-varying size point clouds \( X_i \) obtained via strategy from Section III.A
- \( M_i : \) Number of nonzero elements in each sample \( i \)
- \( M_r : \) upper bound on grid elements to sample
- \( M_{ijc} : \) total number of grid elements in each sample \( i \)

**Result:**
- \( D_j : \) Balanced 4D sampling array

**Algorithm:**
1. Initialize: \( M_r = 2 \cdot \max_{i=1}^{n} M_i, D_j \in \mathbb{R}^{4 \times M_r}, j = 1, \ldots, M_r, c = 0 \)
2. For \( j = 1, \ldots, M_r \)
   - if \( b_j = 1 \)
     - Sample \( M_i \) elements of \( j \) in the \( k = 1 \) nearest neighborhood
   - else
     - Sample \( M_i \) elements of \( j \) in the \( k = 1 \) nearest neighborhood

**End of Algorithm 1**

The intuitive idea of Algorithm 1 is explained as follows: After we have obtained 3D point clouds by measuring a 3D object, we represent the point clouds as voxels. Then we perform kNN sampling selecting points in the vicinity of the surface (green grid elements in Figure 6). Then we obtain a 4D sampling array \( D \) by representing the empty points in the surface vicinity as well as the grid elements with measurement points (red grid elements in Figure 6) by their cartesian coordinate and occupancy status.

This algorithm enables the estimation of meaningful tensor decomposition and significantly reduces the computational effort of evaluating the voluminous binary tensor representation. This is a remedy for the drawbacks of the volumetric and sparse point cloud voxel representation. We note that the input point clouds have a varying number of points \( M_i \), which will lead to a varying number of non-zero entries in the binary tensor \( B_i \). However, to utilize the VAE-based framework introduced in Section III.D, we define a conservative upper bound of \( M_r \) elements to ensure a fixed input dimension to the VAE. To determine \( M_r \), we first select the training sample with the largest number of points from the training
set denoted by $M_{r,max}$. Then $M_r$ is conservatively chosen as $M_r = 1.3 \cdot M_{r,max}$. If the number of measurement points is expected to vary based on the measurement technique, an appropriate upper bound can be selected accordingly. Consequently, when all $M_i$ non-zero entries of the binary tensor $B_i$ are sampled, at most $M_i/2$ non-zero elements will be contained in the binary sampling tensor $D_i$.

C. Streaming Nonlinear Bayesian Tensor Decomposition (SNBTD) via Stream Patch Processing

To take advantage of the superior performance of Bayesian Tensor decompositions on small datasets, we will regularize the VAE embeddings in Section III.D with the low dimensional embeddings $U_{SNBTD}$. Those embeddings are obtained by the SNBTD method, which decomposes the binary sampling tensor $D_i$ that was obtained from the binary point cloud representation $B_i$ of the unstructured, varying-size point cloud samples $X_i$ (Algorithm 1). To make the SNBTD method applicable to the proposed ANTLER framework, several aspects need to be modified. The binary tensors in the proposed framework are extremely sparse (e.g., 0.3% non-zero entries for micro gear dataset) and also very high dimensional (e.g., $1000 \times 1000 \times 1000 = 1$ Billion entries for micro gear dataset). The SNBTD method, on the other hand, is essentially a sparse spectrum approximation of the Gaussian process decomposition model. This sparse spectrum approximation relies on a small set of pseudo inputs, which have a Gaussian distribution prior. For extremely sparse input tensors, this will lead to a bias towards zero, so only zero entries would be reconstructed. To this end, we proposed an efficient sampling strategy in Section III.B (Algorithm 1). This allows us to obtain a balanced dataset representing the 3D shape of the measured objects. As an output of Algorithm 1 (Section III.B), the unstructured point clouds of varying size $X_i$, which were represented as a binary tensor $B_i$, are finally represented as a 4D vector $D_i(i,b_i)$, where $i$ denotes the x,y, and z-coordinates of the binary tensor $B_i$ and the tensor entry value is denoted by $b_i$ (i.e., 0 or 1). Furthermore, we denote a K-mode tensor by $\mathcal{X} \in \mathbb{R}^{d_1 \times \ldots \times d_k}$, where each mode $k$ consists of $d_k$ entries. For decomposition, we introduce $K$ latent embedding matrices $U = \{U^1, \ldots, U^K\}$ to represent the entries in all $K$ modes. Each $U^k$ is $d_k \times r_k$ and the rows are the embedding vectors of the nodes in mode $k$. Furthermore, the entries of the latent embeddings are denoted by $u^k_{ji} = [U^k]_{ji}$. We aim to reconstruct $\mathcal{X}$ (in our case $D$) from $U$.

The workflow of the SNBTD Streaming Patch Processing consists of the following steps:

1) Prior Initialization: The SNBTD method is based on a nonlinear decomposition with random Fourier features. The spectral representation of the stationary kernel function is therefore drawn from $M$ independent frequencies $S = [s_1, \ldots, s_M]^T$. Next, we sample a feature weight vector $w$ from $p(w) = \mathcal{N}(w, 0, M^{-1})$. For details, we refer interested readers to [28]. The rank of the latent embedding $R = \sum_k r_k$. For patch processing, the factorized posterior of the binary tensor

$$q(U,S,w) = \prod_{k=1}^K \prod_{j=1}^{d_k} \prod_{i=1}^{r_k} q(u^k_{ji}) \times \prod_{m=1}^M \prod_{j=1}^R q(s_{mj}) \cdot q(w)$$

is initialized with the prior, where each $q(u^k_{ji}) = \mathcal{N}(u^k_{ji} | \mu^k_{ji}, \sigma^k_{ji})$, $q(s_{mj}) = \mathcal{N}(s_{mj} | a_{mj}, p_{mj})$, $q(w) = \mathcal{N}(w | \eta, \Sigma)$, and the first and second moments as $\mu^k_{ji} = E_p[u^k_{ji}]$, and $\sigma^k_{ji} = E_p[u^k_{ji}] - E_p[u^k_{ji}]^2$, respectively. We refer interested readers to [28] for further details. If some engineering knowledge about the shape of the objects is available beforehand, a more informative prior distribution can be specified.

2) Stream Patch Processing: Patches $P_i$ of the sample $D_i$ arrive during training and need to be processed to update the posterior.

a) Blending distribution: All data points in the current patch are then utilized to construct a blending distribution

$$p_b(\theta_{p_b}) = p_b(\theta_{p_b} | s_m) q(s_{mj}) \prod_{i \in b_i} \psi \left( (2b_i - 1)w^T \phi(x_i) \right),$$

where $\theta_{p_b} = \{U,S,w\}, \theta_{p_b} | s_m = \theta_{p_b} | s_m = \theta_{p_b} | s_m$ and $p_b(s_m | \theta_{p_b} | s_m) \propto q(s_{mj}) \prod_{i \in b_i} \Phi \left( (2b_i - 1)w^T \phi(x_i) \right)$.

b) Parallel update: Next, $\{q(s_{mj})\}$, $\{q(u^k_{ji})\}$ and $q(w)$ is updated in parallel with conditional moment matching coupled with Gaussian-Hermite quadrature and Taylor approximation to ensure computational tractability [28].

c) Updated posterior: Based on the computations in step 2.b, the posterior $q(\cdot)$ from Eq. 2 is updated. This process is repeated until all patches $P_i$ in each sample $D_i$ are processed.

The dimension of the latent embeddings $R$ is selected via binary search to find the smallest possible dimension that achieves a 99.5% reconstruction performance based on Cross-validation.

Another challenge is the high dimensionality of those tensors: to efficiently precompute the low dimensional embeddings $U_{SNBTD}$, which serves as an input to the VAE with ANTLER loss, we adopt the streaming patch processing for the SNBTD. This method allows the processing of observed tensor entries in small patches by updating the posterior of latent embeddings $U_{SNBTD}$, the weights $w$ and the frequencies $S$ based on each patch of the sampling tensor $D_i$, without using the previously accessed patches $[P_1, \ldots, P_{i-1}]$. The process of streaming patch processing of the SNBTD is illustrated in Figure 7.

D. Nonlinear Regression Approach

The goal of the proposed method, contrary to an abundance of previous research in the field of tensor regressions, is to model the relationship between the low dimensional embedding $U$ learned in Section III.C and the response $Y$ with the following nonlinear form:

$$Y = g_0(U) + E,$$
where \( \theta \) are the model parameters of a nonlinear function \( g(\cdot) \) to be estimated, and \( E \) is the error tensor. The proposed framework allows the integration of various off-the-shelf nonlinear regression methods such as random forests or deep neural networks. An appropriate method can be chosen based on cross-validation. This overcomes the extreme shortcoming of previous tensor regression methods [7], [10] that rely both on linear decomposition and linear regression techniques, deeming them impractical in many real-world engineering applications, which exhibit complex nonlinear structures.

For the remainder of this paper, without loss of generality, we will utilize Deep Neural Networks (DNN) to perform the nonlinear regression task. This is due to the high expressiveness of properly trained DNNs and the implicit self-regularization in DNNs derived by Martin et al. [32]. Even though the dimensionality of the binary tensor was significantly reduced due to Algorithm 1, the dimension of the low dimensional embeddings \( \mathcal{U} \) is still relatively high, which considering the small sample size in many engineering applications may lead to overfitting and the curse of dimensionality. Therefore, we recommend utilizing regularization strategies in the estimation of \( \theta \) to alleviate representational difficulties and bad generalization behavior. In particular, DNN training exhibits a Tikhonov-like form of self-regularization, which can be enhanced by explicit forms of regularization such as dropout and weight norm constraints. There is a “size scale” separating signal from noise. Therefore, by exploiting the generalization gap phenomena, smaller DNN (such as the ANTLER architecture) can be regularized by simply changing the batch size. A batch size of 1 will lead to a more regularized model [32]. Following these theoretical results, we recommend a batch size of 1 to achieve the maximal self-regularization in training with small sample sizes.

The optimization of the network architecture is conducted based on common techniques such as grid search based on cross-validation.

### E. Unified ANTLER Framework for Efficient Inference

In this section, we propose an efficient framework to simultaneously optimize the components of the ANTLER framework and achieve a computationally tractable approximation of the nonlinear tensor decomposition introduced in Section III.C and the nonlinear regression method in Section III.D.

The SNBTD has a time complexity of order \( O(N \sum_k d_k R_k + M R + 4M^2) \) [28] which is prohibitive for an online setting. Typical values in the proposed setting are a sample size of \( N = 100 \) with \( d_k = 1,000,000 \) \((k = 1, 2, 3)\) points in each sample, a dimension of each embedding vector \( r_k \) in the range 3-8, \( R = \sum_k R_k \) and \( M = 128 \) independent frequencies. To alleviate the computational burden of the SNBTD, but still exploit its immense nonlinear tensor decomposition capability and simultaneously update the nonlinear tensor decomposition as well as the nonlinear regression, we propose the following novel ANTLER loss function:

\[
L_{\phi, \theta, \theta_0}(D) = \sum_{D \in \mathcal{D}} \log \frac{1}{S} \sum_{j=1}^{S} \rho_0(D_j | z_0) + \lambda_1 D_{KL}(q_\phi(z | D_j) \| p(z)) + \lambda_2 \| \mu_z - \mu_{\theta_0, SNBTD} \|_2^2 \\
+ \lambda_3 \| g_\theta(z | \mu_z) - Y \|_2^2, \tag{4}
\]

where \( \lambda_1, \lambda_2, \lambda_3 \) are tuning parameters, \( \mu_z \) denotes the mean of \( z, \mu_{\theta_0, SNBTD} \) denote the mean of the low dimensional embedding \( U \) obtained via the SNBTD method and the reparameterization trick is applied as \( z = g(e, \phi, D_j) \). The terms in the loss function have the following interpretations:

- The first two terms represent the original ELBO loss function from the classical VAE. The first term denotes the expected error in reconstructing the binary data tensor \( D \) from \( z \). The second term denotes the Kullback-Leibler (KL) divergence, which ensures that auxiliary distribution \( q(z) \) is close in terms of distribution to \( p(z) \).

- The third term explicitly regularizes the solution of VAE to approximate the mean of the low dimensional embedding \( U \) obtained via SNBTD, \( \mu_{\theta_0, SNBTD} \). The additional SNBTD loss alleviates the drawback of VAEs, that priors are often highly redundant due to i.i.d. assumptions on internal parameters [33]. In contrast to vanilla VAEs, this allows the latent embeddings to model the full range of statistical properties of the input data. By enforcing the SNBTD properties, we promote diversity in the latent variables, which is difficult to achieve in the generative framework since VAEs typically learn “incorrect” latent representations [34]. This is important since the salient statistical properties of the low dimensional embedding will be used for discrimination. Embeddings need to span the space of possible data and represent diverse characteristics, which is important for discriminative tasks. However, the SNBTD method is not applied directly since the posterior distribution does not have a closed-form expression. The posterior optimization must rely on variational inference. This is computationally expensive, especially when the number of predictors is large. Therefore, it is necessary to find a more computationally tractable way to represent the SNBTD low-dimensional embeddings. Since a two-layer perceptron can approximate any (nonlinear) function on a bounded region [35], [36], a VAE can be used to approximate the posterior distribution of the SNBTD without the need for the expensive Bayesian inference.

Therefore, we can integrate the advantages of both deep VAEs and Bayesian methods. We note that for \( \lambda_2 = 0 \), this SNBTD property of the VAE disappears. However, our experimental results on the synthetic and gear datasets showed
that including this term achieves significantly higher prediction performance.

The fourth term models the nonlinear relationship between the mean $\mu_x$ of the low dimensional embeddings $z$ and the response $Y$. By integrating this directly into the VAE, we can simultaneously estimate the nonlinear tensor decomposition and nonlinear regression in a one-step approach to alleviate the suboptimality of a two-step approach. Contrary to two-step approaches, this one-step approach enables the recognition of variation patterns that are correlated with the binary input tensor as verified in the simulation studies.

Consequently, the ANTLER algorithm is given as follows.

**Algorithm 1**

**Part 1: Binary Tensor Representation**

Represent the unstructured varying-size point cloud samples

$$X_i \in \mathbb{R}^{3 \times M_i} \text{ as binary tensors } B_i \in \mathbb{R}^{d_i \times d_i \times d_i} \text{ (Section IIIA)}$$

**Part 2: Efficient Balanced Sampling Method for Extremely Sparse Tensors**

Obtain $D \in \mathbb{R}^{4 \times M} \times \mathbb{R}$ from $B$ utilizing Algorithm 1

**Part 3: Streaming Nonlinear Bayesian Tensor Decomposition (SNBTD)**

Via Streaming patch processing

During training: Obtain low-dimensional SNBTD embeddings $\Lambda_{\text{SNBTD}}$ (Section III.C) to serve as a regularizer for Part 4

**Part 4: VAE with ANTLER loss**

$(\phi, \theta, \theta_r) \leftarrow$ Initialize parameters

while SGD is not converged do

$$M \sim D \text{ (Random minibatch of data)}$$

$$\epsilon \sim p(\epsilon) \text{ (Random noise for every data point in } M \text{ (Reparameterization trick))}$$

Compute $L_{\phi, \theta, \theta_r}(M, \epsilon)$ from Eq. 4 and its gradients $\nabla_{\phi, \theta, \theta_r} L_{\phi, \theta, \theta_r}(M, \epsilon)$

Update $\phi$, $\theta$ and $\theta_r$ using SGD optimizer

end

**IV. SIMULATION STUDIES**

In this section, we evaluate the ANTLER approach with simulated varying-size unstructured 3D point clouds against three benchmark methods. The common data characteristics are as follows: (i) the parts are measured and represented by unstructured point clouds of varying size, and (ii) the goal is to develop a model to link the point cloud inputs with a scalar or vector response. In the simulation studies, we will use two guiding examples to illustrate these characteristics: In the example of waveform surfaces the surface roughness serves as a response. For the conic shapes, we predict the roundness error.

Following Yan et al. [7], we consider a wave-shaped surface and truncated cones to generate point cloud objects. We simulated $N$ structured point clouds as training samples $X_i$, $i = 1, \ldots, N$ with $M$ measurement points. The simulated data are...
A. Wave-Shape Surface Point Cloud Simulation

Surface point clouds have important applications in surface prediction for precision manufacturing. Therefore, we simulate surface point clouds in a 3D Cartesian coordinate system, where \( 0 \leq x, y \leq 1 \). The corresponding \( z_i \) values at \( \left( \frac{u_i}{r}, \frac{v_i}{r} \right) \), \( i = 1, \ldots, 12 \), are determined by \( X_i \) for the \( i \)-th sample. The surface point clouds are generated according to \( X_i \) and result in areas with different point densities. The vector regression response \( Y_i \) is computed by applying a nonlinear function \( f(\cdot) \) on the unstructured point clouds \( X_i \) of varying size (i.e., \( Y_i = f_r(X_i \cdot \cdot) \)).

Fig. 8. Examples of generated wave-shape surfaces.

B. Truncated Cone Point Cloud Simulation

In manufacturing applications, conic shapes are commonly used as a reference object for problems such as part-to-part variation pattern identification [45] or process control [7]. Therefore, following Yan et al. [7], we simulate truncated cone point clouds in a 3D cylindrical coordinate system \((r, \phi, z)\), where \( \phi \in [0, 2\pi] \) and \( z \in [0, 1] \). The corresponding \( r \) values at \((\phi, z) = \left( \frac{2\pi}{n}, \frac{z}{r_0} \right) \), \( i_1 = 1, \ldots, i_2, i_2 = 1, \ldots, i_2 \) with \( i_1 = i_2 = 1,000 \) for the \( i \)-th sample are recorded in the matrix \( X_i \). The vector regression response \( Y_i \) is generated according to \( V = B \times X \) where \( B \in \mathbb{R}^{3 \times 3} \times 2 \) are generated as

\[
B_1 = \begin{pmatrix} 4 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \quad \text{and} \quad B_2 = \begin{pmatrix} 1 & 3 & 0 \\ 1 & 0 & 2 \end{pmatrix} .
\]

Furthermore, three basis matrices \( U^{(k)} = [u_1^{(k)}, u_2^{(k)}] \) are selected with \( u_\alpha^{(k)} = \sin(\frac{\pi z}{r_0}), \sin(\frac{2\pi x}{a}), \ldots, \sin(\frac{2\pi z}{a}) \), \( \alpha = 1, 2, 3 \). The input matrix \( Z \) is sampled from a normal distribution \( \mathcal{N}(0, 1) \). Three examples of the generated surface point clouds are shown in Fig. 8.

An important application of 3D point clouds is the parameterization of surface roughness based on microwave remote sensing [42]. The obtained surface roughness is an important indicator in the modeling of surface dynamics, such as soil erosion or runoff estimation [42], [43]. Therefore, we compute the surface roughness \( R_a \) based on point cloud data to serve as a regression response. First, a reference plane is defined as \( z = b_0 + b_1 x + b_2 y \), where \( x, y \), and \( z \) are spatial coordinates in \( \mathbb{R}^3 \) and the vector of plane coefficients \( \beta = [b_0, b_1, b_2] \) is obtained via orthogonal distance regression. In particular, \( \beta = (X^T X - \delta^2 I)^{-1} X^T z \), where \( X \) is the design matrix of \( x \) and \( y \) coordinates and \( \delta \) is the smallest singular value of the augmented matrix \([X, z] \). By scaling the normal vector \( n \) to unit length, the orthogonal point-to-plane distance \( D_j \) is computed as the positive inner product of the observation vector \( v_j \) and the unit normal vector \( n_a \) as \( D_j = n_a \cdot v_j \). Finally, the surface roughness response is calculated as \( R_{\text{line}} = Y_i = \frac{1}{m_e} \sum_{j=1}^{m_e} (D_j - D)^2 \) [44], which is a nonlinear function of the input point cloud \( X_i \).

Fig. 9. Examples of generated truncated cones.
Finally, the average roundness $\bar{R}(x, y)$ over all circumferential lines is used as the response $Y_i$.

C. Benchmarks and Data Preprocessing

In this section, we will compare the proposed ANTLER framework with three existing methods.

From the field of tensor regressions, we will use the Multiple Tensor on Tensor (MTOT) regression [10] as a benchmark. To make these methods applicable for unstructured point cloud data, we will use the binary sampling tensor $D$ obtained from the raw measurement point cloud in order to make the data structured and applicable to the MTOT method. We note, that this method heavily favors the MTOT method since it utilizes the efficient representation and Algorithm 1 of our proposed framework. The MTOT framework by itself could not handle the unstructured, varying-size 3D point cloud inputs directly.

As a nonlinear tensor regression method, we utilize Tensor Regression Networks (TRN) [12]. Since the raw point clouds are not permutation invariant and the binary sampling tensor $B$ was too sparse to produce meaningful results, similarly to the MTOT method we use the binary sampling tensor $D$ as an input to this method.

As a deep learning benchmark, we will use a modified PointNet++ architecture [25]. This method was originally designed for classification tasks but can be modified for regression tasks by adjusting the output layers activation function from a normalized exponential (Softmax) to a Rectified Linear Unit (ReLU) function. In the remainder of this paper, this method is referred to as PointNet Regression (PointNetR++).

From the area of graph convolution networks (GCN) for point cloud processing, we compare against the Grid-GCN [26], which was modified from its classification model to regression tasks (Grid-GCN-R).

RepSurf-U [27] is the currently best performing method for single point cloud inputs on ModelNet40. This model uses a novel representation, that explicitly learns local structures in point clouds, which is the most related to our regression task. We only modify it from classification to regression tasks (RepSurf-U-R).

Furthermore, we will use a classical machine learning benchmark to evaluate our proposed algorithm. Previous research by Biehler [49] has compared multiple machine learning methods for regression tasks with unstructured point cloud input and concluded that Random Forest (RF) outperforms all other benchmarks. Therefore, we will only use the RF as a benchmark method in the simulation and case study. The features are extracted inspired by the “key to the PointNet approach” [1], which uses a symmetric function (e.g., maximum) to aggregate the information from point clouds. In particular, a min and max function was applied to obtain the 10,000 largest and smallest points of each coordinate axis. Several other ways of feature extraction such as covariance analysis of the distance-driven local neighborhoods [50], Riemann graphs over local neighborhoods [51], multiscale and hierarchical point clusters [52], and a self-organizing network for point cloud analysis [53] were also investigated but yielded inferior results.

For 3D measurement acquisition devices, the number of measurement points in different samples usually does not vary by order of magnitudes. Since we chose the lower bound of points as the number for subsampling, the resulting fixed-size point clouds usually contain at least half of the original measurement points. Therefore, the choice of subsampling technique does not significantly influence the results. However, more advanced subsampling techniques, such as important support points [54] to find an “optimal” subsample from weighted proposal samples, could be applied under extreme measurement conditions.

D. Simulation Study Prediction Results

For each method, we compare the proposed method with benchmark methods based on the Root Means Squared Error (RMSE) calculated at different levels of noise $\delta$ and different sample sizes $N$. Table I reports the average and standard deviation of RMSE obtained via 10-fold Cross Validation for the simulation cases 1 (Waveform Surface) and 2 (Truncated Cone), respectively.

In all cases, the proposed ANTLER framework achieves state-of-the-art performance, reflecting the advantage of the ANTLER method in terms of prediction accuracy. The performance improvement is mainly due to the following three reasons: (i) an efficient tensor representation of the point clouds, (ii) a nonlinear Bayesian tensor decomposition, and (iii) nonlinear regression models integrated by a unified VAE framework. Furthermore, with an increase in $\delta$, all methods exhibit a larger RMSE and variance of the prediction results.

As the sample size increases, the deep learning-based methods improve and the SNBTD regularization of the ANTLER method is less effective than on small datasets ($N = 100$ to $N = 1,000$). On small to medium-sized datasets, our joint ANTLER framework for decomposition and regression enables the learning of small local features, most correlated with the regression response. The superior performance on small sample sizes is further enabled through the Bayesian SNBTD regularization of the VAE-based ANTLER framework. When the sample size increases the advantage through regularization that promotes diversity in the latent space diminishes.

The other benchmark methods cannot outperform the proposed ANTLER due to their own limitations as follows: (i) Even with significant help from our proposed point cloud representation and Algorithm 1, the MTOT method and nonlinear tensor regression methods such as TRN cannot handle the unstructured nature of the point clouds effectively and relies on the linear assumption. Therefore, it fails to capture the complex, nonlinear relationships between unstructured point clouds and a vector response. (ii) The deep learning methods (PointNet++, Grid-GCN-R, RepSurf-U-R) fail to efficiently represent the high dimensional input data as a low dimension
TABLE I
RMSEs and Standard Deviation (SD) for Cases 1 and 2
(Note: bold indicates superior method)

| Method          | Case 1: Waveform Surface | Method          | Case 2: Truncated Cone |
|-----------------|--------------------------|-----------------|------------------------|
| $\delta = 0.1$  | $N=100$                  | $\delta = 0.01$ | $N=100$                |
| RMSE            | 18.832                   | RMSE            | 31.784                 |
| SD              | 4.687                    | SD              | 9.273                  |
| MTOT            |                          | MTOT            |                        |
| PointNetR++     | 13.446                   | PointNetR++     | 25.906                 |
| Grid-GCN-R      | 11.442                   | Grid-GCN-R      | 16.402                 |
| RepSurf-U-R     | 10.382                   | RepSurf-U-R     | 11.574                 |
| TRN             | 10.382                   | TRN             | 13.391                 |
| **ANTLER**      | **6.768**                | **ANTLER**      | **9.874**              |
| $\delta = 0.1$  | $N=1,000$                | $\delta = 0.01$ | $N=1,000$              |
| RMSE            | 38.048                   | RMSE            | 43.929                 |
| SD              | 10.102                   | SD              | 6.289                  |
| MTOT            |                          | MTOT            |                        |
| PointNetR++     | 37.852                   | PointNetR++     | 31.706                 |
| RF              | 34.109                   | RF              | 34.883                 |
| Grid-GCN-R      | 29.487                   | Grid-GCN-R      | 29.269                 |
| RepSurf-U-R     | 29.487                   | RepSurf-U-R     | 21.978                 |
| TRN             | 22.959                   | TRN             | 28.472                 |
| **ANTLER**      | **14.297**               | **ANTLER**      | **21.372**             |
| $\delta = 0.1$  | $N=10,000$               | $\delta = 0.01$ | $N=10,000$             |
| RMSE            | 14.991                   | RMSE            | 14.638                 |
| SD              | 5.523                    | SD              | 4.193                  |
| MTOT            |                          | MTOT            |                        |
| PointNetR++     | 12.578                   | PointNetR++     | 13.426                 |
| RF              | 13.912                   | RF              | 14.927                 |
| Grid-GCN-R      | 7.885                    | Grid-GCN-R      | 9.195                  |
| RepSurf-U-R     | 5.569                    | RepSurf-U-R     | 8.186                  |
| TRN             | 6.744                    | TRN             | 9.517                  |
| **ANTLER**      | **4.397**                | **ANTLER**      | **6.971**              |
| $\delta = 0.1$  | $N=100$                  | $\delta = 0.01$ | $N=100$                |
| RMSE            | 13.929                   | RMSE            | 14.638                 |
| SD              | 2.464                    | SD              | 4.193                  |
| MTOT            |                          | MTOT            |                        |
| PointNetR++     | 4.556                    | PointNetR++     | 9.145                  |
| RF              | 5.215                    | RF              | 10.953                 |
| Grid-GCN-R      | 4.222                    | Grid-GCN-R      | 6.678                  |
| RepSurf-U-R     | 3.747                    | RepSurf-U-R     | 6.057                  |
| TRN             | 3.9533                   | TRN             | 6.232                  |
| **ANTLER**      | **3.473**                | **ANTLER**      | **6.064**              |

Fig. 10. Gear measurement point clouds.

ANTLER methodology. Due to the high engineering tolerance requirements, numerous manufacturing defects occur, causing increased costs. However, in the current state of the art, the quality of the final product, such as gear boxes, can only be ensured through an end-of-line test bench or extensive FEA simulations [55]. However, predicting the functional characteristics of gears during cycle time is the cornerstone of selective assembly and the optimization of gear noise emissions. This is of particular interest in applications such as automotive or medical engineering.

To advance the state of the art and achieve in-process quality improvements (IPQI), the gears need to be measured during the cycle time. Subsequently, based on the obtained point clouds, the functional characteristics of interest need to be predicted. By identifying defective parts in terms of their engineering tolerance and function fulfillment, a more holistic approach toward the identification of scrap and selective assembly can be achieved.

V. CASE STUDY

In this section, a real case study concerning gears manufactured by micro gear hobbing is introduced and used as a guiding example for the use of the proposed manifold for small sample sizes. Due to the deep learning architecture, a very large number of parameters needs to be estimated from a limited number of samples leading to overfitting. Additionally, none of the architectures was intentionally designed for regression tasks. Only the RepSurf-U method explicitly tries to learn from local information, which explains its comparable performance on medium datasets ($N=10,000$).

(iii) The Random Forest does not exploit the rich spatial information contained in point clouds by using a two-step approach, which does not consider the regression response during feature extraction. Additionally, the feature extraction fails to capture small local features highly correlated with the regression response.

A. Gear Dataset

The case study is based on 120 optically measured gears as described in preliminary work by Gauder et al. [56], [57]. The gears were measured in a climate-controlled measuring room on an Alicona $\mu$CMM. The $\mu$CMM is a highly accurate, purely optical, non-contact 3D micro-coordinate measurement machine and enables the measurement of dimensions, position, shape, and surface roughness. The results of the measurement are three-dimensional point clouds, as shown in Fig. 10.

Due to the manual clamping, the orientation and rotation of the measurement point clouds may not be aligned. To align the point clouds, the CAD model is used as a reference, and the rotation and translation matrix of the point clouds is obtained via the Iterative Closest Point (ICP) algorithm [58]. After the initial experiments, where the clamping process was conducted manually, the clamping of the gears in the measurement device will be automated, and the process of alignment via ICP can be skipped. To obtain the transmission error and eccentricity in both directions of rotation, the gear-specific FEA software ZaKo3D is utilized [59]. Firstly, based on the gear measurements (i.e., point clouds) and a master gear representing the ideal gear design, finite element structures are generated. Then the simulation applies a nonlinear finite element model, which has been calibrated with real-world testbench experiments. For details on this procedure for this case study, we refer interested readers to Gauder et al. [60].

Several outputs such as transmission error, loads, and deflects on the tooth are computed. The FEA computation time for one sample is approximately 2 hours, so much
longer than the cycle time of approximately 3 minutes For further background on the nonlinearity of the simulation outputs, readers are referred to [61], [62]. In the remainder of this paper, the four response variables for the regression framework are the following: the clockwise transmission error (TE) and eccentricity (E) are denoted by \( Y^1 \) and \( Y^2 \), while the counterclockwise TE and E are denoted by \( Y^3 \) and \( Y^4 \), respectively.

While the different predictive models have different computational times during training time, the inference time during testing is the determining factor for our application. We have reported the inference times for the case study in Table II. The corresponding hardware consists of a personal computer with an Intel Core i7-7600U and 16GB RAM.

All deep learning methods (including our VAE-based framework) enjoy fast inference time during the forward pass through the network for a given test sample. MTOT is the only method not based on deep learning. Therefore, it is an outlier in terms of inference time since it needs to solve an optimization problem at test time.

The main factor for the inference time is the pre-processing of the raw 3D point clouds. For the deep learning benchmarks (PointNet++, Grid-GCN, and Rep-Surf-U), the point cloud data needs to be up- and down-sampled to a fixed size to deal with the varying size of measurement points in each sample. For our ANTLER method and the tensor regression methods (MTOT and TRN) the data needs to be represented as voxels and Algorithm 1 needs to be applied to obtain the input for the models. For the RF we extract min- and max features as described in Section IV C.

The cycle time in the gear application is 4 minutes and therefore, all methods (besides MTOT) satisfy in-process prediction requirements.

During the processing of the raw point clouds, the origin and z-axis orientation are assumed to be fixed for each point cloud. For the alignment of point clouds, we propose two strategies depending on the measurement setup: In controlled environments, where the measurement objects can be approximately aligned before each measurement, the point clouds can be used directly. If no alignment of subsequent samples can be guaranteed, the ICP method can be adopted for alignment of the point clouds.

### Table II

| Method     | Preprocessing Method | Preprocessing Time [s] | Model Inference Time [s] | Total Test Time [s] |
|------------|----------------------|------------------------|-------------------------|---------------------|
| MTOT       | Voxelization Algorithm 1 | 183.58                 | 597.68                  | 1171.26             |
| PointNet++ | Sampling to fixed size | 79.245                 | 0.002                   | 79.247              |
| RF         | Extraction of features | 52.546                 | 0.001                   | 52.547              |
| Grid-GCN-R | Sampling to fixed size | 79.245                 | 0.002                   | 79.247              |
| RepSurf-U-R| Sampling to fixed size | 79.245                 | 0.002                   | 79.247              |
| TRN        | Voxelization Algorithm 1 | 183.58                 | 0.001                   | 183.582             |
| ANTLER     | Voxelization Algorithm 1 | 183.58                 | 0.003                   | 183.584             |

Fig. 11. Boxplots of RMSEs for predicting the functional response using our ANTLER framework and six benchmarks.

### B. Case Study Prediction Results

To develop a predictive model, perform 10-fold cross-validation (CV). In this paper, we compared the same six benchmark methods as the simulation study, random forest (RF), Multiple Tensor-on-tensor regression (MTOT), Tensor regression networks (TRN), and the deep learning methods PointNetR++, Grid-GCN-R, RepSurf-U-R. Fig. 11 depicts the boxplot of root mean squared errors (RMSE) obtained over 120 measurements for the proposed method and the benchmarks.

The results show the superiority of the proposed ANTLER framework in comparison to six benchmark methods. On average, across all CV scores and response variables, the RSME of the ANTLER method is 258.5%, 285.6%, 562.8%, 424.7%, 308.1%, and 282.55% smaller than the respective benchmark methods MTOT, RF PointNetR++, Grid-GCN-R, RepSurf-U-R, and TRN, respectively. The deep learning methods exhibit a high variance due to the small sample sizes and complexity, and the nonlinear structure of the learning task, which leads to overfitting. The tensor regression methods MTOT and TRN enjoy a significant advantage by using our sampling tensor \( D \) obtained by Algorithm 1 as an input. ANTLER predicts the vector response more accurately, capturing the complex nonlinear relationship by exploiting the information in unstructured point clouds of varying size in a simultaneous and disciplined manner. The results make a

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on August 31,2024 at 15:58:52 UTC from IEEE Xplore. Restrictions apply.
case for point cloud modeling in complex, nonlinear regression settings.

VI. Conclusion

Discriminative models for unstructured point cloud data are an emerging research area, with applications in smart manufacturing, geological surveying, or autonomous driving due to the capabilities of 3D scanners in accurate shape modeling. To address an important industry problem and several research gaps in the processing of unstructured, varying-size point clouds, we proposed a novel ANTLER framework. This approach directly uses the entire information contained in the high-dimensional unstructured point cloud with varying sizes, structures it as a binary tensor, utilizes a balanced sampling strategy, and further employs efficient algorithms for simultaneous tensor decomposition and nonlinear regression. Two simulation studies and a real-world case study illustrate the effectiveness of the proposed approach for in-process functional prediction based on an unstructured 3D point cloud input of varying sizes. Future work could extend this framework to the detection and classification of multiple types of surface anomalies. Moreover, the application of the high-level idea to other tensor learning problems (e.g., nonlinear multiple tensors on tensor regression) is a promising direction. Finally, tensor completion or robust tensor estimation techniques could be applied to explicitly handle missing values or noise in the input point clouds.

References

[1] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, “PointNet: Deep learning on point sets for 3D classification and segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 652–660.

[2] J. Du, H. Yan, T.-S. Chang, and J. Shi, “A tensor voting-based surface anomaly classification approach by using 3D point cloud data,” J. Manuf. Sci. Eng., vol. 144, no. 5, May 2022, Art. no. 051005.

[3] C. Li, C. H. Fu, Y. B. Guo, and F. Z. Fang, “A multiscale modeling approach for fast prediction of part distortion in selective laser melting,” J. Mater. Process. Technol., vol. 229, pp. 703–712, Mar. 2016.

[4] J.-P. Vincent, C. Baudouin, J. Bruyere, J.-Y. Dantan, and R. Bigot, “Gear metrology of statistical tolerancing by numerical simulation,” in Advanced Mathematical and Computational Tools in Metrology and Testing. Singapore: World Scientific, 2009, pp. 344–350.

[5] T. Fukuchi and T. Shiba, Industrial Applications of Laser Remote Sensing. Bentham Science Publishers, 2012.

[6] A. Filgueira, H. González-Jorge, S. Lagiela, L. Díaz-Vilarinho, and P. Arias, “Quantifying the influence of rain in LiDAR performance,” Measurement, vol. 95, pp. 143–148, Jan. 2017.

[7] H. Yan, K. Paynabar, and M. Pacella, “Structured point cloud data analysis via regularized tensor regression for process modeling and optimization,” Technometrics, vol. 61, no. 3, pp. 385–395, Jul. 2019.

[8] W. Shi and R. Rajkumar, “Point-GNN: Graph neural network for 3D object detection in a point cloud,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 1711–1719.

[9] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, “Deep learning for 3D point clouds: A survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 12, pp. 4338–4364, Dec. 2020.

[10] M. R. Gahrooei, H. Yan, K. Paynabar, and J. Shi, “Multiple tensor-on-tensor regression: An approach for modeling processes with heterogeneous sources of data,” Technometrics, vol. 63, no. 2, pp. 147–159, Apr. 2021.

[11] F. Wang, M. R. Gahrooei, Z. Zhong, T. Tang, and J. Shi, “An augmented regression model for tensors with missing values,” IEEE Trans. Autom. Sci. Eng., vol. 19, no. 4, pp. 2968–2984, Oct. 2020.

[12] J. Kossaifi, Z. C. Lipton, A. Kolbeinsson, A. Khanna, T. Furlanello, and A. Anandkumar, “Tensor regression networks,” J. Mach. Learn. Res., vol. 21, no. 1, pp. 4862–4882, Jan. 2020.
[40] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian optimization of machine learning algorithms,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012.

[41] D. R. Jones, “A taxonomy of global optimization methods based on response surfaces,” *J. Global Optim.*, vol. 21, no. 4, pp. 345–383, 2001.

[42] S. Gharechelou, R. Tateishi, and B. A. Johnson, “A simple method for the parameterization of surface roughness from microwave remote sensing,” *Remote Sens.*, vol. 10, no. 11, p. 1711, Oct. 2018.

[43] M. J. M. Römkens, K. Helming, and S. N. Prasad, “Soil erosion under different rainfall intensities, surface roughness, and soil water regimes,” *CATENA*, vol. 46, nos. 2–3, pp. 103–123, Jan. 2002.

[44] R. M. Polleyea and J. P. Fairley, “Estimating surface roughness of terrestrial laser scan data using orthogonal distance regression,” *Geology*, vol. 39, no. 7, pp. 623–626, Jul. 2011.

[45] A. T. Bui and D. W. Apley, “Analyzing nonparametric part-to-part variation in surface point cloud data,” *Technometrics*, pp. 1–18, 2021.

[46] H. Miao, A. Wang, B. Li, T.-S. Chang, and J. Shi, “Process modeling with multi-level categorical inputs via variable selection and level aggregation,” *IIE Trans.*, pp. 1–14, Jan. 2021.

[47] K. Dransfield, K. Addinall, and P. Bills, “Comparison and appraisal of techniques for the determination of material loss from tapered orthopaedic surfaces,” *Wear*, vol. 478–479, Aug. 2021, Art. no. 203903.

[48] G. Moroni and S. Petro, “Geometric tolerance evaluation: A discussion on minimum zone fitting algorithms,” *Precis. Eng.*, vol. 32, no. 3, pp. 232–237, Jul. 2008.

[49] M. Biehler, “AI-based prediction of functional characteristics of micro gears and uncertainty quantification,” M.S. thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany, 2021.

[50] M. Pauly, R. Keiser, and M. Gross, “Multi-scale feature extraction from point-set surfaces,” *Comput. Graph. Forum*, vol. 22, no. 3, pp. 281–289, Sep. 2003.

[51] S. Gumbhold, X. Wang, and R. S. MacLeod, “Feature extraction from point clouds,” in *Proc. IMR*, 2001, pp. 293–305.

[52] Z. Wang et al., “A multiscale and hierarchical feature extraction method for terrestrial laser scanning point cloud classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2409–2425, 2014.

[53] J. Li, B. M. Chen, and G. H. Lee, “SO-Net: Self-organizing network for point cloud analysis,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9397–9406.

[54] C. Huang, V. R. Joseph, and S. Mak, “Population quasi-Monte Carlo,” 2020, arXiv:2012.13769.

[55] S. Park, S. Kim, and J.-H. Choi, “Gear fault diagnosis using transmission error and ensemble empirical mode decomposition,” *Mech. Syst. Signal Process.*, vol. 108, pp. 262–275, Aug. 2018.

[56] D. Gauder, J. Götz, M. Biehler, M. Diener, and G. Lanza, “Balancing the trade-off between measurement uncertainty and measurement time in optical metrology using design of experiments, meta-modelling and convex programming,” *CIRP J. Manuf. Sci. Technol.*, vol. 35, pp. 209–216, Nov. 2021.

[57] D. Gauder, R. Wagner, J. Götz, B. Häfner, and G. Lanza, “Funktionsorientierte qualitätssicherung von mikrozahnrädern hinsichtlich des geräuschverhaltens,” *Technisches Messen*, vol. 86, no. 9, pp. 469–477, 2019.

[58] P. J. Besl and N. D. McKay, “Method for registration of 3-D shapes,” *Proc. SPIE*, vol. 1611, pp. 586–606, Apr. 1992.

[59] C. Brecher et al., “Zak03D-simulation possibilities for PM gears,” in *Proc. World Congr. Powder Metal*, Florenz, Italy, 2010.

[60] D. Gauder, J. Götz, A. Bott, and G. Lanza, “Comparative analysis between single flank rolling test for micro gears and analytical simulation of optical measurements,” *Technisches Messen*, vol. 89, no. 9, pp. 580–593, Sep. 2022.

[61] J. Ma, C. Li, and L. Cui, “Transmission error analysis and disturbance optimization of two-stage spur gear space driven mechanism with large inertia load,” *Shock Vibrat.*, vol. 2018, pp. 1–14, Jun. 2018.

[62] L. Oliveri, C. Rosso, and S. Zucca, “Influence of actual static transmission error and contact ratio on gear engagement dynamics,” in *Nonlinear Dynamics*, vol. 1. Cham, Switzerland: Springer, 2017, pp. 143–154.

Michael Biehler

Member, IEEE

received the B.S. and M.S. degrees in industrial engineering and management information systems from the Karlsruhe Institute of Technology (KIT) in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. His research rests at the interface between machine learning and cyber-physical (manufacturing) systems, where he aims to develop methods for cyber security, monitoring, prognostics, and control. He is a member of ASA, ASME, ENBIS, IIE, INFORMS, and SIAM.

Hao Yan

received the B.S. degree in physics from Peking University, Beijing, China, in 2011, and the M.S. degree in statistics, the M.S. degree in computational science and engineering, and the Ph.D. degree in industrial engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2015, 2016, and 2017, respectively. He is currently an Assistant Professor with the School of Computing, Informatics, and Decision Systems Engineering (SCIDSE), Arizona State University (ASU), Tempe, AZ, USA. His research interests focus on developing scalable statistical learning algorithms for large-scale high-dimensional data with complex heterogeneous structures to extract useful information for the purpose of system performance assessment, anomaly detection, intelligent sampling, and decision making. He is a member of INFORMS and IIE. He was a recipient of multiple awards, including the Best Paper Award in the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING and the ASQ Brumbaugh Award.

Jianjun Shi

received the B.S. and M.S. degrees in automation from the Beijing Institute of Technology in 1984 and 1987, respectively, and the Ph.D. degree in mechanical engineering from the University of Michigan in 1992. Currently, he is the Carolyn J. Stewart Chair and a Professor at the Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. His research interests include the fusion of advanced statistical and domain knowledge to develop methodologies for modeling, monitoring, diagnosis, and control of complex manufacturing systems, and is a fellow of four professional societies, including ASME, IIE, INFORMS, and SME, an Elected Member of the International Statistics Institute (ISI), a Life Member of ASA, an Academician of the International Academy for Quality (IAQ), and a member of the National Academy of Engineers (NAE).