www.cadanda.com Computer-Aided Design AND APPLICATIONS

Invariance Class based Surface Reconstruction using CNN Transfer Learning

Yifan Qie[1] , Nabil Anwer[2]

[1]Université Paris-Saclay, ENS Paris-Saclay, LURPA, yifan.qie@ens-paris-saclay.fr
[2]Université Paris-Saclay, ENS Paris-Saclay, LURPA, nabil.anwer@ens-paris-saclay.fr

Corresponding author: Yifan Qie, yifan.qie@ens-paris-saclay.fr

Abstract. Geometrical operations are defined in ISO standards on geometrical product specifications and verification (GPS) for obtaining ideal and non-ideal features to represent surfaces on mechanical parts. These geometric features should be categorized by their geometrical properties for further geometry processing. In the context of ISO GPS, surface portions are determined by their kinematic invariance and are classified into planar, cylindrical, helical, spherical, revolute, prismatic and complex surfaces. In this paper, geometric features based on discrete representation enabled by the Skin Model Shapes paradigm are investigated. An automatic approach for surface reconstruction by invariance class is proposed in this paper based on deep learning architecture to speed up geometry processing. Partitioning operation is implemented to decompose a mechanical part from simulations or measurements into surface portions. Regarding the specificity of the derived point cloud, a view-based strategy is implemented by dimension reduction using point cloud projection. The extracted view-wise features are aggregated into a discriminative global representation for training neural networks. The method is tested on an open dataset and a case study is given for validation. The proposed method enables robust surface reconstruction for reverse engineering with low computational cost in the context of ISO GPS.

Keywords: Invariance Class, Point Cloud, Skin Model Shapes, Surface Reconstruction, Geometry Processing, Transfer Learning, Reverse Engineering
DOI: https://doi.org/10.14733/cadaps.2023.1204-1220

1 INTRODUCTION

Geometry processing has been widely investigated and applied in computer-aided design with the evolution of simulation technologies as well as 3D data acquisition techniques. In ISO standards on geometrical product specifications and verification (GPS) [13], geometric features are obtained by geometrical operations to represent functional and non-functional surface portions on mechanical parts. These geometric features are determined by their kinematic invariance and are classified into planar, cylindrical, helical, spherical, revolute, prismatic and complex surfaces in the context of ISO GPS for further geometry processing .

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

The geometry of a manufactured part differs from the ideal part created using CAD systems due to the geometric variations. In ISO GPS, the concept of Skin Model is proposed to build the bridge between CAD and physical objects by defining the physical interface between a workpiece and its environment [13]. To enable the operability of the Skin Model concept for the use in geometric variations, Skin Model Shapes are proposed to represent geometric features in discrete geometry [30].

Many research activities for Reverse Engineering (RE) have contributed to invariance class identification and surface reconstruction of mechanical parts from non-ideal representations such as Skin Model Shapes. Partitioning point clouds into homogeneous regions based on different criteria is often implemented since it enables decomposing of the input point clouds of full mechanical parts into individual surface portions. Edge detection methods [43, 3], region growing methods [18, 4], attributes clustering methods [39, 17], Spectral analysis methods [41, 38] are able to decompose a point cloud into regions that share similar properties. Further surface reconstruction steps are needed for these methods since they are not able to identify surface types automatically. Shape fitting methods, on the other hand, try to fit primitive shapes such as planes, cylinders and spheres to the point clouds [8, 2, 26, 36]. The partitioned surface portions are obtained by their primitive types. The primitive shapes defined in each shape fitting methods are based on different mathematical classifications and they are not able to cover all the invariance classes defined in ISO GPS. Comprehensive surveys have been conducted in [32, 35, 28] with details of these methods.

Hybrid methods benefit from the aforementioned approaches and can be applied for surface reconstruction in the context of ISO GPS. Gelfand et al. proposed a method based on slippage analysis that is able to identify surface patches based on points' normal information [10]. A probabilistic description by pre-defined semi-parametric models [7] also permitted to enrich the work of invariant surface identification. Schnabel proposed Random Sample Consensus (RANSAC) based method to detect surface primitives in unorganized point clouds [31]. The work can be used for single surface reconstruction with a pre-defined primitive set. Cai et al. proposed a three-step hybrid process towards ISO GPS for partitioning mesh and point cloud into surface portions and identifying each of them as one of the seven invariance classes of surfaces [6]. Qie et al. simplified the three step method and improved the performance by considering the natural boundaries on the partitioned surface [25]. The aforementioned methods are able to address the invariant surface identification and reconstruction by either pre-defined parameters or manually-established models. However, the performances of these methods are limited due to the need of prior knowledge and the high computational complexity.

In this paper, the implementation of Convolutional Neural Network (CNN) techniques is conducted to speed up the geometry processing on the discrete representation enabled by the Skin Model Shapes paradigm. Partitioned point clouds are processed for invariance identification using fine-tuned AlexNet by transfer learning. Ideal surfaces are obtained based on the approximation for each point cloud according to their invariances.

The main contributions of this paper includes:

a) A novel method is proposed for automatic point cloud partitioning;

b) The deep learning techniques are implemented for surface reconstruction from the discrete representation;

c) An optimal reverse engineering framework is introduced by eliminating the numerical errors in the process of constructing meshes and estimating the normal of vertices.

The remainder of this paper is organized as follows: Related work is introduced in section 2. The framework of the proposed surface reconstruction is proposed in section 3. The partitioning operation using a local point attribute called surface variation is introduced in section 4. The surface reconstruction by transfer learning is provided in section 5. Finally, experiments on a CAD object from an online benchmark are illustrated to highlight the performance of the proposed method.

2 LITERATURE REVIEW

As a well-known deep learning algorithm in computer vision to tackle problems such as image classification, face recognition, etc., CNN has been widely investigated and applied in mechanical engineering due to its effectiveness in capturing the input data features. Nevertheless, the implementation of CNN on point cloud for CAD is hindered due to the sparsity, randomness and non-structure of point clouds. Existing deep learning strategies for point cloud identification and classification [12, 42] can be categorized into multi-view based methods, volumetric based methods and point based methods.

Since the point cloud shows characteristics of sparsity, randomness and non-structure, indirect ways such as multi-view based methods and volumetric based methods convert such data into a regular structure, and then process the structure with networks. Multi-views based methods transform 3D data into 2D and then apply existing techniques to extract knowledge. Su et al. proposed a multi-view convolutional neural network (MVCNN) to complete classification of 3D point cloud. The methods tried to project 3D point cloud into 2D images and implement CNN to extract features from the images and identify the object [33]. Boulch et al. put forward a deep segmentation networks to semantic label unstructured point cloud [5]. They compute dense labeling in the multi-view-based images and back project the result of the semantic segmentation to the original point cloud. SnapNet-R is proposed by [11] to provide solution of simpler image and geometry integration as compared to SnapNet. Volumetric methods transform unstructured point clouds into the regular volumetric occupancy grid. Then the regular representation of the point cloud could be used as the input of the learning network to train the model and achieve segmentation. VoxNet is proposed as an architecture to integrate a volumetric occupancy grid representation with a supervised 3D Convolutional Neural Network (3D CNN) [20]. The appearance of the voxelized objects depends heavily on the origin, orientation and resolution of the voxel grid in space. Tchapmi el al. proposed a SEGCloud to process voxelized 3D point cloud that combines the machine learning techniques (Conditional Random Field, CRF) and deep learning methods (3D CNN) [34].

Unlike indirect ways of point cloud that convert such data into a regular structure, direct ways are developed in the context of deep CNN which only use coordinate of the point could as the input of neural networks. It is proposed to make full use of the characteristics of the data and reduce the computational complexity. Qi et al. proposed a PointNet [22] to achieve effective feature learning. The proposed architecture of the learning network enables to support classification, part segmentation and sematic segmentation with different output layer configurations. Based on the work of PointNet, researches are conducted in order to improve the performance of the algorithm [42, 27]. PointCNN are proposed to deal with the disorder of point clouds [19]. X-Conv operator is built in the model to retain features with the order of input points. A-CNN is able to be implemented to achieve semantic segmentation of large scenes with annular convolution neural network [16]. Wang et al proposed a Graph CNN (GCNN) to realize the semantic segmentation of point cloud by operates directly on the graph structure [37]. Friedrich et al. proposed a hybrid approach for segmentation and fitting primitives to 3D point clouds. The method adapted PointNet++ to label points on point could with associated primitive type including plane, cylinder and sphere and post processing are conducted with approaches such as DBSCAN, RNASAC, and Genetic Algorithms [9]. Yan et al. put forward HPNet for primitive segmentation using hybrid representations. The work considers descriptors derived from predicted geometric parameters to improve the final performance on CAD models [40].

Given an extra data dimension, 3D data such as point clouds has many advantages for representing mechanical parts compared to 2D and image data. Meanwhile, the large number of labeled data required for training a CNN can be achieved with relatively little effort by geometry processing on point clouds rather than proposing a novel theory for 2D and image data such as Generative adversarial networks (GANs) [14]. Therefore, it is promising to implement deep learning techniques on 3D point clouds to speed up geometry processing for CAD models. Zhu et al. put forward a Deep learning learning based method for geometric deviation modeling in Additive Manufacturing (AM) [44]. The in-plane surface deformation are modelled and mapped into n deviation space for a learning process. Qie proposed a data-driven methods for non-default partitioning [23] that uses the PointNet architecture for identifying compound features from CAD point clouds. The identification of such features enables the further operations in the context of ISO GPS.

3 METHOD OVERVIEW

An automatic algorithm for surface reconstruction by its invariance class is proposed based on deep learning architecture to speed up the geometry processing in this paper. Discrete geometric features enabled by the Skin Model Shapes paradigm are investigated in this paper. The discrete representations of mechanical parts are based on point cloud obtained by tessellation on either nominal model or Skin Model Shapes or by measurements of manufactured parts.

The framework of the proposed surface reconstruction method on point cloud is shown in Fig. 1. Point clouds obtained from simulations or measurements are decomposed into surface portions by the partitioning operation. In this paper, a novel method for point cloud partitioning using a point attribute called surface variance [21] is introduced. The step is followed by a invariance identification step for recognizing the surface types for each partitioned point cloud. In order to overcome the sparsity, randomness and non-structure of the point clouds scattered in 3D space, a pre-processing step of dimension reduction is conducted by the projection within the principle directions. A deep neural network is trained with labeled point cloud and is used to address the identification step. Finally, the geometric descriptors of each point cloud can be obtained by surface reconstruction since the ideal geometrical features are guaranteed by a fitting step.

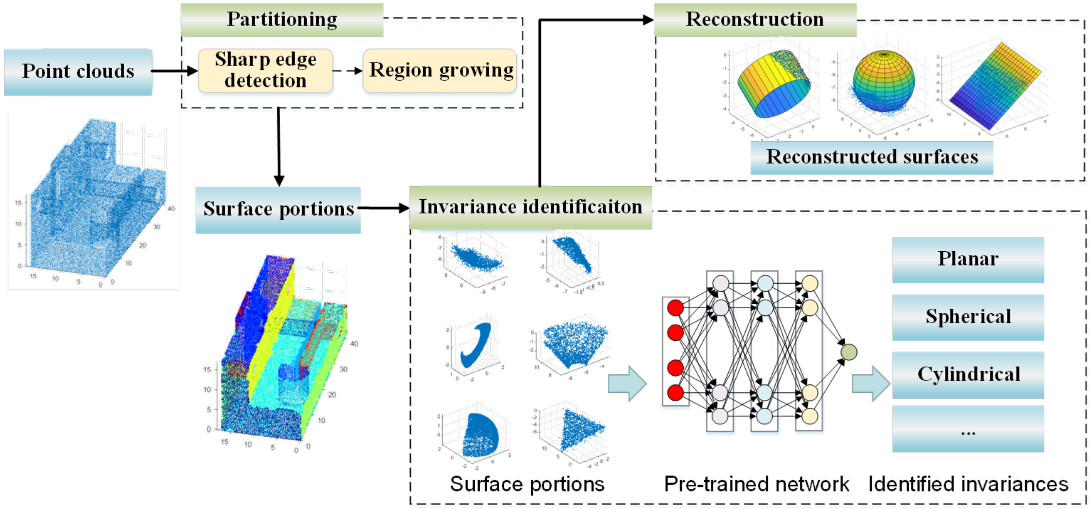


Figure 1: The framework of the proposed two-step surface reconstruction method.

4 PARTITIONING USING SURFACE VARIATION

Partitioning is a fundamental operation that decomposes a mechanical part into independent surface portions for further geometry processing. Research activities have been conducted for mesh partitioning in the context of ISO GPS, such as the curvature-based methods proposed in [25] and [6]. However, the performance of curvature-based methods is limited by the constructed meshes and the estimated normal for each point.

In this paper, a novel approach for ISO GPS partitioning is developed using only the coordinates of points. The approach is based on the observation that the natural boundaries among features on mechanical parts are in most cases edges where an abrupt change of point differential geometry properties occurs. The process of the proposed novel approach is shown in Fig. 2. By first locating the sharp edge on the parts and then group the points inside the boundaries, surface portions are decomposed from the input point cloud. The obtained surface portions will be used for the following reconstruction operation to derive ideal surfaces for each single surface.

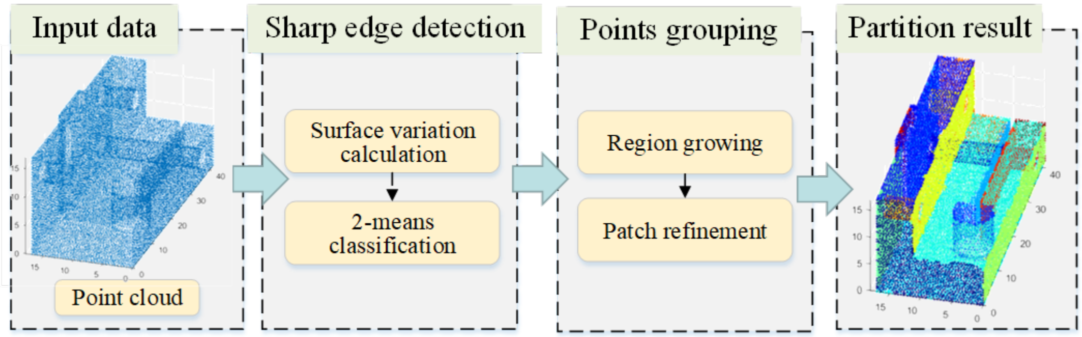


Figure 2: Illustration of the partitioning operation using surface variation.

4.1 Sharp Edge Detection Using Surface Variation

The initial processing of the input point cloud is an edge detection method to locate the points on the edges. While there are a number of techniques for detecting sharp edges, a point attribute called surface variation [21] is considered in this paper to address the detection step since the calculation requires no tessellation of the underlying surface or normal estimations. It performs well to identify the boundaries between surfaces in the point cloud for further operations.

Regarding a point $\mathbf{p}$ in a neighborhood of size n, the surface variation of the point is defined as:

$$\sigma_n(p) = \frac{\lambda_0}{\lambda_0 + \lambda_1 + \lambda_2} \tag{1}$$

where the λ_0, λ_1 and λ_2 are the eigenvalues of a covariance matrix $\mathbf{C}$ for the sample point $\mathbf{p}$ which is defined as:

$$\mathbf{C} = \begin{bmatrix} \mathbf{p}_{i_1} - \overline{\mathbf{p}} \\ \dots \\ \mathbf{p}_{i_1} - \overline{\mathbf{p}} \end{bmatrix}^T \cdot \begin{bmatrix} \mathbf{p}_{i_1} - \overline{\mathbf{p}} \\ \dots \\ \mathbf{p}_{i_1} - \overline{\mathbf{p}} \end{bmatrix}, i_j \in N_p \tag{2}$$

where $\overline{\mathbf{p}}$ is the centroid of the neighbors of $\mathbf{p}$ and it satisfies

$$\mathbf{C} \cdot \mathbf{v}_l = \lambda_l \cdot \mathbf{v}_l, l \in \{0, 1, 2\} \tag{3}$$

since $\mathbf{C}$ is symmetric and positive semi-define.

All the eigenvalues λ_l are real values that measure the variation of the point $\mathbf{p}$ along the direction of the corresponding eigenvectors $\mathbf{v}_l$. λ_0 quantitatively describes the variation along the surface normal. If $\sigma_n(p)$

equals to 0, all the neighbor points lie in a plane. The maximum surface variation $\sigma_n(p) = 1/3$ is assumed for completely isotropically distributed points. The surface variation depends on the size of neighborhood and it is closely related to curvature. Examples of surface variation calculated based on the input point cloud tessellated from CAD objects can be found in Fig. 3.

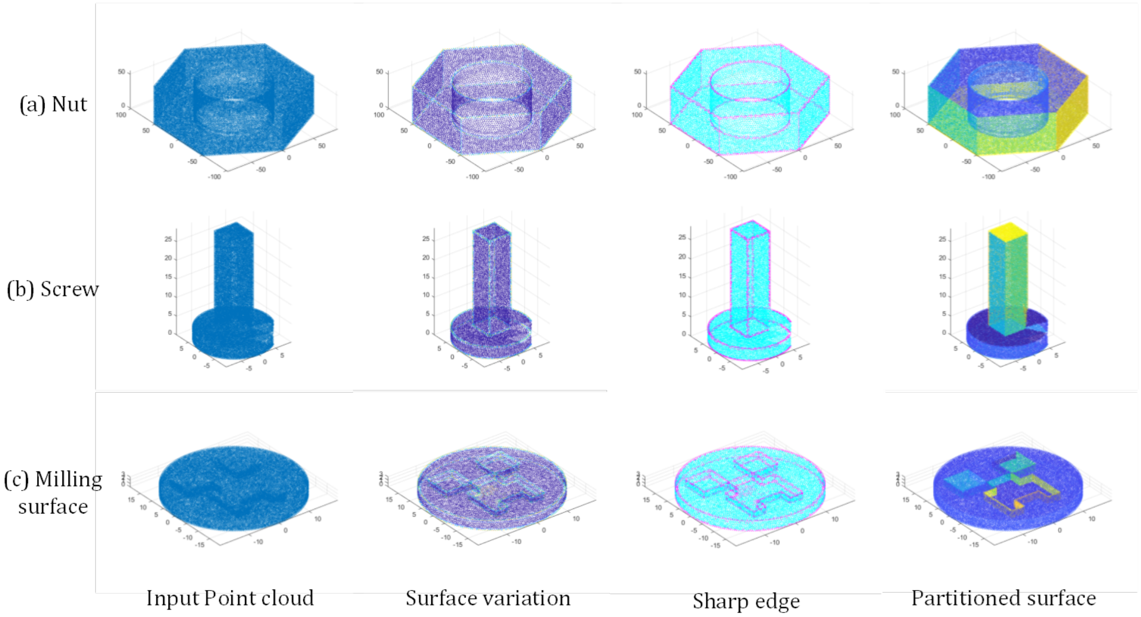


Figure 3: Examples of partitioning operation using surface variation. (Input Point clouds from [29]).

Note that surface variation is a positive value that specifies the amount of the curvatures of the surface. For CAD models, they usually consist of low variation surfaces (i.e. $\sigma_n(p)$ close to 0) and sharp edges ($\sigma_n(p) > 0$ but the values are uncertain regarding input point clouds). Therefore, a classification based on surface variation for each point cloud is realized by a k-means method where k equals to 2. All the points on the sharp edge will locate relatively far from the 0 while low variation surface points locate relatively close to the 0. Examples of sharp edges detected on different input point cloud are illustrated in Fig. 3. Based on the 2-means method, it can be seen that the sharp edge points (points in red) and the non-sharp edge points (points in blue) are automatically and correctly classified. The proposed 2-means points classification for sharp edge detection improves the partition performance by avoiding the pre-defined threshold such as in [6] and [25]. However, it should be highlighted that when the natural boundaries are non-sharp edges where no abrupt change of point differential geometry properties occurs, the performance of the introduced surface variation methods is limited and the consideration of other shape descriptors such as Shape Index and curvedness is suggested as our previous work [25].

4.2 Points Grouping based on the Detected Sharp Edges

Region growing methods for classification rely on the criterion used to decide if the new element such as a point or a triangular face can be added to the existing clustering. In our case, as the sharp edge points are detected on the input point cloud, a region growing method is conducted using the sharp edge label of each point. Table 1 outlines the growing procedure.

Algorithm 1 Region growing on the input point cloud

```
Require: $n \geq 0$                                          ▷ Size of the point cloud
  $N \leftarrow 1$                                              ▷ Group number
  $Group\_Label \leftarrow zeros(n)$                 ▷ Label record for all the points
  for each unlabelled non-sharp edge point $p_i$ do
      queue A $\leftarrow \varnothing$                         ▷ Breadth-First Searching
      $Group\_Label(i) \leftarrow N$
      A.$push(p_i)$
      while $A \neq \varnothing$ do
          $p_j \leftarrow$ A.$pop$
          for each $p_j$ do
              repeat
                  if a neighbor non-sharp edge point $p_k$ is unlabelled then
                      A.$push(p_k)$
                      $Group\_Label(i) \leftarrow N$
                  end if
              until KNN neighbour points of $p_j$ are visited
          end for
      end whileState $N \leftarrow N + 1$
  end for
  return $Group\_Label$
```

Table 1: Pseudo-code for the region growing based on sharp edge points.

Starting from a random point that is labelled as non-sharp type, its neighbors are detected using the KNN approach ($k = 10$). All its non-sharp edge point neighbors are labelled and the procedure repeats by using these neighbors as the new starting points. The growing step stops when all the point within the sharp edge are labelled by the same group number.

The surface portions can be obtained according to the labelled group numbers. However, the partitioned results might be unstable when the point cloud is perturbed or deformed locally. The limitation of the proposed method comes from the choice of the number for the local neighbor points. When the point cloud are non-uniformly sampled, it is possible to have unreasonable partitioning results. Therefore, patch refinement step is considered at the end of the partitioning operation. Methods proposed in [29] can be used in our case as post-processing for surface portions connected by non-sharp boundaries. The surface portions with a relatively small number of points as well as the sharp edge points are merged to their neighbor surface patches to obtain final partitioned surfaces as in Fig 3.

5 SURFACE RECONSTRUCTION BASED ON INVARIANCE CLASS

The input point cloud is partitioned into point cloud subsets representing surface portions of mechanical parts. In order to reconstruct surfaces from the point cloud subset, three consecutive steps including point cloud pre-processing, shape invariance identification and parameter regression, are implemented in this stage. The overview of the proposed surface reconstruction method on point cloud is shown in Fig. 4.

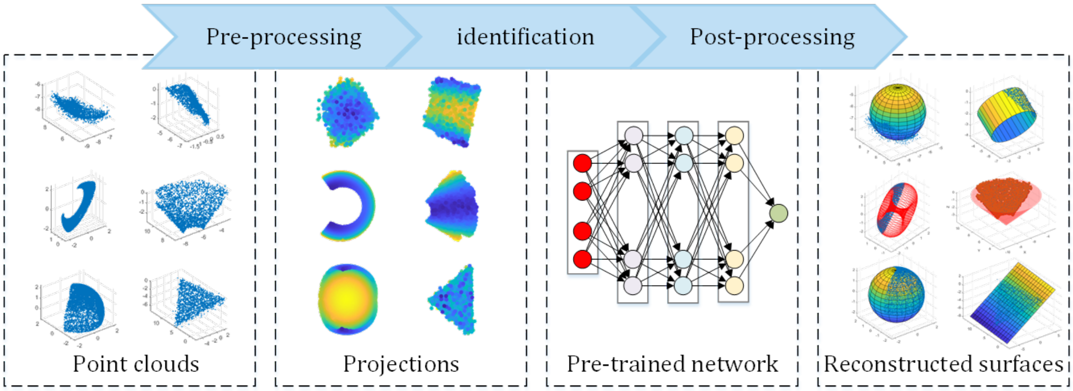


Figure 4: Overview of the proposed surface reconstruction method based on invariance class identification.

5.1 Generating 2D Projections from Point Clouds

Regarding the specificity of the point cloud derived from simulations and measurements, a view-based strategy is implemented by dimension reduction using point cloud projection. The extracted view-wise features are aggregated into a discriminative global representation for further training on neural networks.

Geometrical surfaces are derived from either the Skin Model Shapes in the simulation process or the measurements in the reverse engineering process. The reconstructed information of such surfaces includes both the actual shape (e.g. surface type, contour, etc.) and dimensions as well as geometric deviations of the surface. In early engineering design stages, point clouds are tessellated from CAD models where deviations might be incorporated considering the simulation purposes. During later design stages, manufacturing process simulations and even the prototypes of mechanical parts are available. Point clouds can thus be obtained from the "observation" [30] in these stages with geometric uncertainties.

In this context, the representation of point clouds for geometry processing should be feasible and robust with low computational cost for both nominal surfaces and non-ideal surfaces. The observed non-ideal surfaces can be affected by noise or other imperfections, such as non-uniform sampling, low sampling density, misalignment and missing data, etc. Therefore, as a statistical method to compute the most meaningful basis to re-express the point sets, Principal Components Analysis (PCA) is implemented in our method as pre-processing approach for extracting the principal axes of point clouds.

Considering a discrete point cloud partitioned from a workpiece, the covariance matrix is defined as:

$$M_{cov} = \sum_{i=1}^{n} (x_i - O_{pca})(x_i - O_{pca})^T \tag{4}$$

where O_{pca} is the origin determined as the centroid of the points of the point cloud. The first principal axis is the eigenvector corresponding to the largest eigenvalue. The two other principal axes are obtained from the remaining eigenvectors.

Based on the analysis, the projection process starts with the normalization of point clouds into a unit sphere, which addresses the problem of sparsity and randomness of the 3D data. Then the normalized point cloud is mapped along the first principal axis. A filtering process is conducted in the mapping process to overcome the sparsity by eliminating unnecessary details of the mapping patterns. In the end, the mapping patterns are transferred into images to address the 'non-structure' problem of the point clouds. However, the mapping along the first principal axis is not enough for surface invariance class identification due to the missing information of the altitude of each point along the first principal axis. Therefore, the altitude along the first principal axis is transferred into specified colors during the mapping process to maintain the information. Examples can be found in Fig. 5 with the point cloud mapping results of different types of invariance classes. It can be seen from the figure that some of the information such as the edges of the surface and the noises are filtered out during projection process, which is the major cost in the training process of point-based deep learning methods such as PointNet[22].

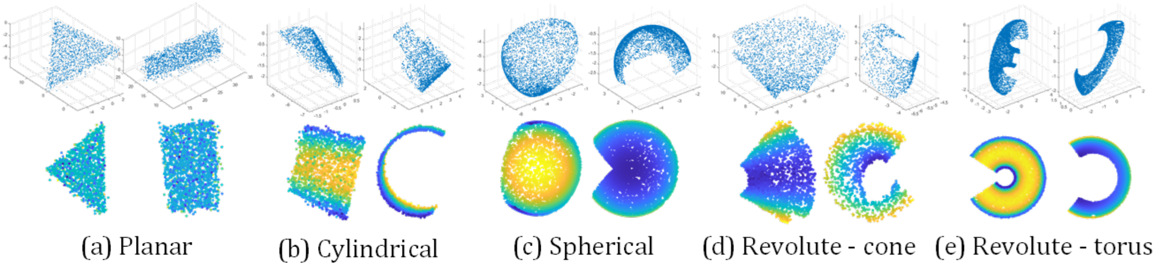


Figure 5: The projection of 3D point clouds to 2D structure image data, classified by their invariance classes. The first row of in the figure is the original point cloud and the second row is the mapped image of each point cloud.

5.2 Transfer Learning for Surface Identification

The objective of transfer learning is to transfer the knowledge from the existing task to a new target task. It is widely applied regarding the limited labeled examples or the significant amount of training time in the new tasks. Considering the pre-trained CNNs contain generic and reusable features in the first few layers that are directly related to our problem, transfer learning on CNN is implemented in this paper.

As a large network structure with 60 million parameters and 650,000 neurons, the pre-trained AlexNet on the ImageNet data set is employed in this paper for transfer learning due to its performance for image classification tasks. The the architecture of AlexNet is given in Table 2. It contains 11 layers and an input layer and an output layer. The output layer of the pre-trained AlexNet has 1000 classification scores since the data in the ImageNet is classified into 1000 categories.

The structures of AlexNet and the TransferNet are compared in Figure 6. For invariant surface identification using the fine-tuned TransferNet, all the layers, except for these last three layers, are extracted from AlexNet. The updated last three layers of the TransferNet consist of a fully connected layer, a soft-max layer, and a classification output layer. The fully connected layer is set to have the same size as the invariance classes in ISO GPS (here we set it as 5 instead of 7 in practice). In the transfer learning process, the parameters of all the layers of the pre-trained AlexNet, except for these last three layers, are kept and these layers are called frozen layers. The parameters in the last layers will be trained using our own dataset to fine tune the performance for surface invariance identification.

In order to verify the performance of our method, the experiment is conducted by transfer learning on an open-source benchmark dataset from SHREC'22 (http://shrec.ge.imati.cnr.it/shrec22_fitting/). It is a large-scale dataset with 3D segments represented as point clouds, some of the representatives of the data set are shown in Figure 7. Surface primitives include planes, cylinders, spheres, cones and torus that are classified by their invariance classes. 46,000 point clouds are contained in the dataset as the training set and 925 point clouds are used as the test set. The point clouds could be clean or perturbed by noises following a variety of distributions.

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	image	1	227x227x3	-	-	-
1	Convolution	96	55x55x96	11x11	4	Relu
2	Max pooling	96	27x27x96	3x3	2	Relu
3	Convolution	256	27x27x256	5x5	1	Relu
4	Max pooling	256	13x13x256	3x3	2	Relu
5	Convolution	384	13x13x384	3x3	1	Relu
6	Convolution	384	13x13x384	3x3	1	Relu
7	Convolution	256	13x13x256	3x3	1	Relu
8	Max pooling	256	6x6x256	3x3	2	Relu
9	Fully connected	-	9216	-	-	Relu
10	Fully connected	-	4096	-	-	Relu
11	Fully connected	-	4096	-	-	Relu
Output	Fully connected	-	1000	-	-	Softamx

Table 2: The architecture of AlexNet.

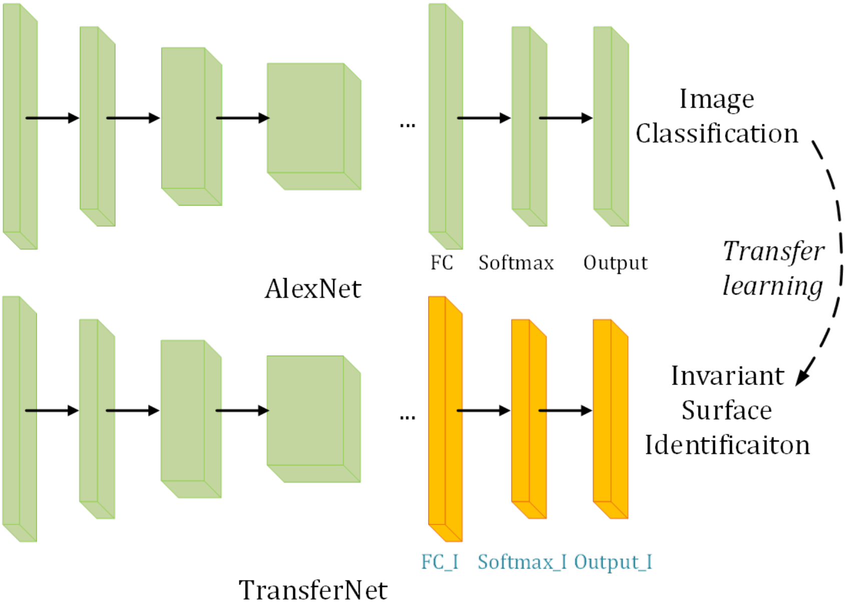


Figure 6: Illustration of transfer learning from AlexNet.

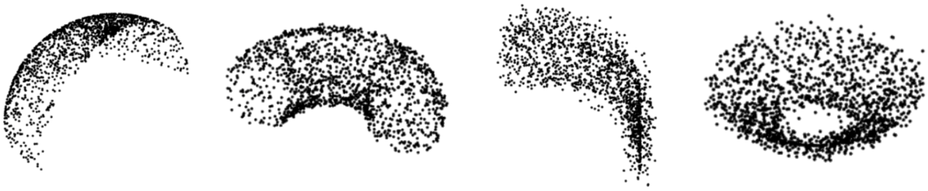
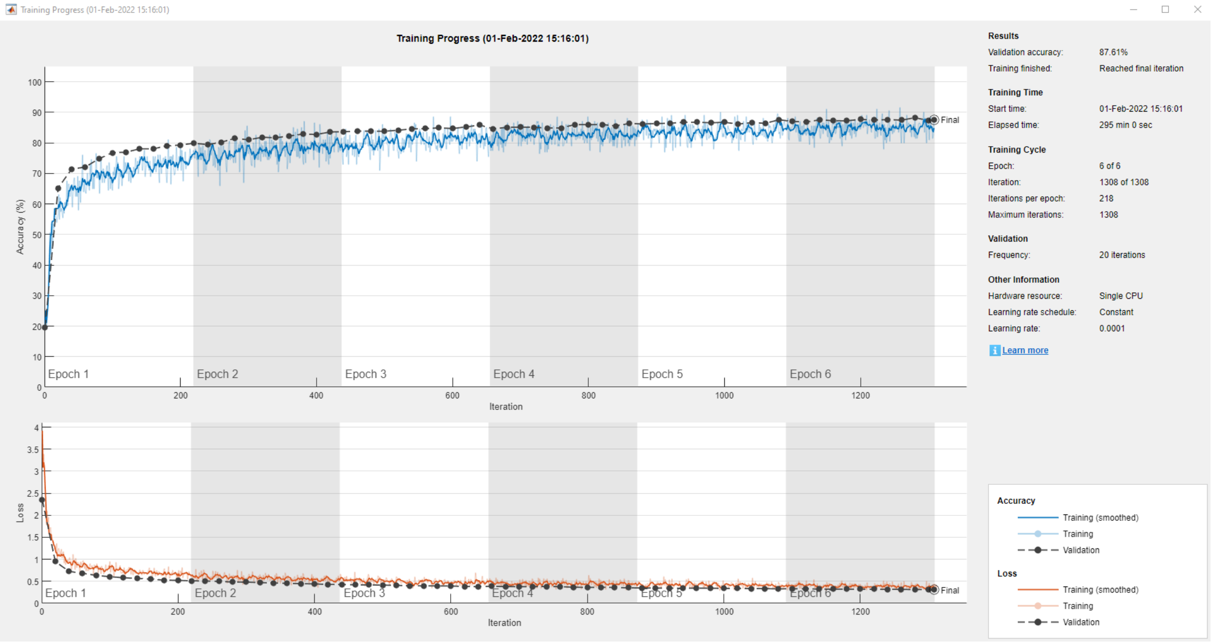


Figure 7: Some examples in SHREC'22 track on the fitting and recognition of simple geometric primitives on point clouds (http://shrec.ge.imati.cnr.it/shrec22_fitting/).

The training is performed on a workstation equipped with a 1.70 GHz Intel(R) Xeon(R) CPU, 16 GB RAM, and the Windows 10 operating system. The learning curve is shown in Figure 8. Our approach took 5h to train the AlexNet for the SHREC'22 dataset (minibatch size 256 for 6 epochs). The validation accuracy reached 87.61% after training.



Training Progress (01-Feb-2022 15:16:01)

Results	
Validation accuracy:	87.61%
Training finished:	Reached final iteration
Training Time	
Start time:	01-Feb-2022 15:16:01
Elapsed time:	295 min 0 sec
Training Cycle	
Epoch:	6 of 6
Iteration:	1308 of 1308
Iterations per epoch:	218
Maximum iterations:	1308
Validation	
Frequency:	20 iterations
Other Information	
Hardware resource:	Single CPU
Learning rate schedule:	Constant
Learning rate:	0.0001

Figure 8: The training process of the fine-tuned AlexNet.

5.3 Surface Reconstruction by Parameter Regression

In the context of ISO GPS, the association operation is defined to fit the pre-defined features to non-ideal point clouds considering specified criteria including Gaussian (least squares), Chebyshev and maximum inscribed or minimum circumscribed fittings. Details about surface reconstruction towards ISO GPS can be found in [24]. When the invariance class of a point cloud is identified by the fine-tuned AlexNet, an ideal feature defined by a equation can be guaranteed to obtain a reconstructed surface for reverse engineering.

To improve the performance of the proposed method on the evaluated dataset, a $geometry - validation$ strategy is implemented to deal with bias from the fine-tuned AlexNet. After Analyzing the confusion matrices over the whole test set, it is noticed that the cones and tori in the test set are most difficult to be correctly recognized and they are likely to be classified as cylindrical invariance. Therefore, a quick validation for cylindrical surfaces based on geometry property is conducted to determine if the assumed cylindrical invariance is reasonable. Considering the distributions of surface variation introduced in Section 4 are different for cylinders, cones, and tori in the test set, the initial cylindrical classification guess is assumed wrong if the distribution of surface variation exceeds a pre-defined threshold.

Other geometry properties such as normal distribution, Gaussian curvature, mean curvature, curvedness and shape index [25] can also be considered as a post-processing step to improve the performance of surface reconstruction results. The utilization of such geometry properties depends on the input specificity (point could, uniform Voronoi region, and 2-manifold meshes, etc.) and computational requirements. Finally, the distances between the points in point cloud and the reconstructed surface are measured to obtain an optimal reconstruction results.

Different approaches from the literature review are compared in this section for their performances regarding a variety of variations point cloud specificity as summarized in Table 3. 12 types of point cloud are considered in the table, for example, the first type (No.1) represents the point clouds that are clean and uniformly sampled. The last type (No.12) represents the point could that have Gaussian random noise and contain missing parts on the surface. The approaches, including slippage analysis based method [10], statistical modeling based method [7], RANSAC based method [31] and a hybrid method [6], are considered in the table. The point clouds could be clean or perturbed by systematic and random deviations as considered by Skin Model Shapes paradigms. Meanwhile the properties of point clouds are different due to the sampling strategies such as uniform sampling, under sampling and sampling with missing parts.

Our method is able to deal with different types of point cloud specificity while other methods are limited by their performance. Extra information, such as normal directions and local neighbors of each point, and pre-defined parameters are not needed by our method. We show that our method, while simple and effective, is robust to various kinds of input corruptions. It should be highlighted that the computational time of our method for each point cloud is less than 0.5 seconds on average while the other methods take several minutes to obtain the surface type information.

6 CASE STUDY

The implementation of our method is examined in this section with a case study. Point clouds representing CAD objects are obtained from a benchmark called Fit4CAD and tested by our method. The dataset consists of 225 high quality point clouds, each of which has been obtained by sampling a CAD object [29].

One of the point clouds in Fit4CAD benchmark is visualized in Figure 9 to illustrate the feasibility of the proposed method. A button-head screw object is tessellated into a point cloud of 17917 points. The point cloud is clean but affected by non-uniform sampling.

Partitioning operation is implemented on the point could by first detecting the vertices on the sharp edges using surface variation. The surface portions partitioned from the point could are visualized by different colors as shown in Figure 9 (b). The fine-tuned AlexNet is used to identify each surface portions represented as point clouds and classify the surfaces by their invariance types. In Figure 9 (c) and (d), we highlight the surfaces on the parts by different colors according to their invariance. Here the planar surfaces are plotted in red, cylindrical surfaces in green and spherical surfaces cyan.

Surface reconstruction based on invariance class is then conducted by fitting ideal planes (cylinders, spheres) to detected planar (cylindrical,spherical) surfaces, respectively. Some of the ideal surfaces associated with the partitioned surface portions are shown in Figure 9 (e), (f) and (g). In order to illustrate the accuracy and

No.	Deviation [1]	Sampling[2]	Ref [10]	Ref [7]	Ref [31]	Ref [6]	Ours
1	C	Uni	●	●	●	●	●
2	C	Und		●	●	●	●
3	C	M	●	●	●		●
4	S	Uni	●	●	●	●	●
5	S	Und		●	●		●
6	S	M	●	●	●		●
7	U	Uni	●	●		●	●
8	U	Und					●
9	U	M	●	●			●
10	G	Uni	●	●	●	●	●
11	G	Und			●		●
12	G	M	●	●	●		●
Remark	-	-	*	**	***	****	

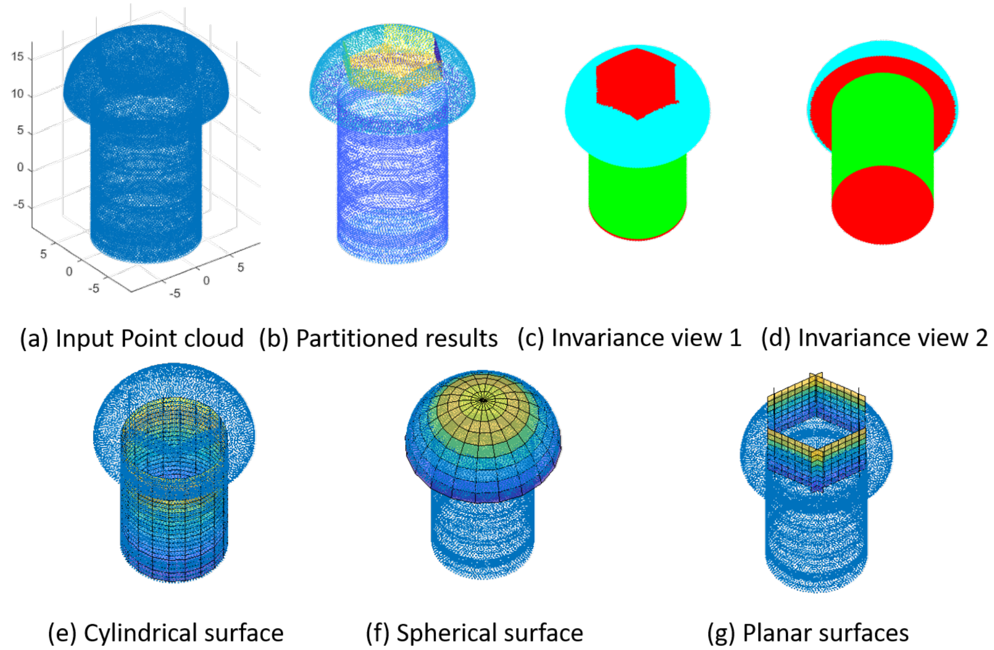
[1] C: Clean; S: Systematic deviations; U: Uniform random deviations; G: Gaussian random deviations;
[2] Uni: Uniform; Und: Under-sampling; M: Missing;
* Normal information needed; Parameter needs to be tuned for different noise deviations.
** Pre-defined statistical models; High computation complexity.
*** Pre-defined models; Parameter needs to be tuned for different noise deviations.
**** Normal information needed; Parameter needs to be tuned for different noise deviations.

Table 3: Performance of different approaches regarding point cloud specificity.

efficiency of our methods, more case are provided in Figure 10. The models are obtained from the ABC dataset that contains a big number of mechanical shapes [15].The identified surface invariance from the shapes are also labelled by different colors. The surfaces are correctly partitioned and the surface invariance is identified by our method efficiently. The key characteristics of each invariance class, such as the normal vector of planar surface or the radius as well as the rotational axis of a cylindrical surface, are obtained. These obtained characteristics can be referred for the following 3D CAD modelling step in reverse engineering to create a geometric (solid) product model [1].

7 CONCLUSIONS

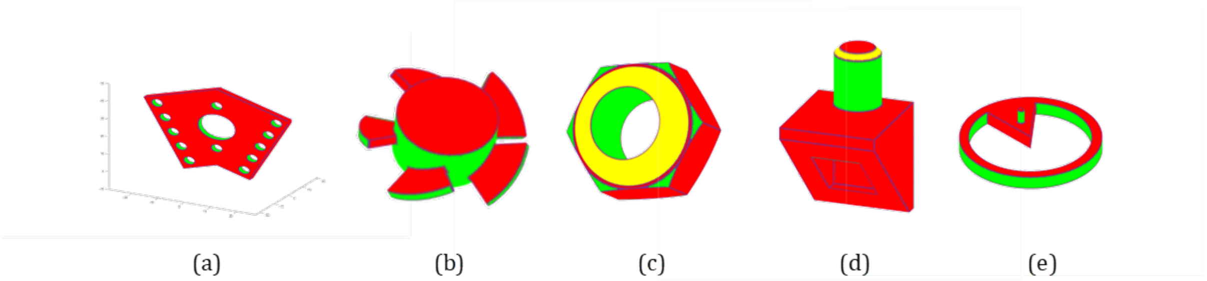
In this paper, a method based on deep learning is proposed for surface reconstruction based on surface invariance classification defined in ISO GPS. Point clouds tessellated from Skin Model Shapes or obtained from measurements are considered as input. Local parameters such as normal information and global parameters such as threshold for sharp edge detection are not required by the proposed method. The input point cloud is partitioned by a 2-means approach based on surface variation and then each surface portion is identified by its invariance using a fine-tuned AlexNet. Pre-processing and post-processing are implanted to improve the performance of surface reconstruction. the The method is tested on an open dataset and it enables robust surface reconstruction for reverse engineering with little computational cost in the context of ISO GPS. A case study is introduced to illustrate the implementation of the proposed method. Future work will focus on the establishment of key indicators for estimating the reconstruction performance to optimise product development phases.



(a) Input Point cloud (b) Partitioned results (c) Invariance view 1 (d) Invariance view 2

(e) Cylindrical surface (f) Spherical surface (g) Planar surfaces

Figure 9: Case study of the proposed method on the point cloud from Fit4CAD benchmark [29].



(a) (b) (c) (d) (e)

Figure 10: Additional examples for testing the method (The indicated colors are associated with the surface invariance class. red : planar ; green : cylindrical ; yellow : rotation).

ACKNOWLEDGEMENTS

This research has benefited from the financial support of China Scholarship Council (first author), under Grant NO.201806020187.

Yifan Qie, http://orcid.org/0000-0002-0347-6077
Nabil Anwer, http://orcid.org/0000-0002-0771-4685

REFERENCES

[1] Anwer, N.; Mathieu, L.: From reverse engineering to shape engineering in mechanical design. CIRP Annals, 65(1), 165–168, 2016. http://doi.org/10.1016/j.cirp.2016.04.052.

[2] Attene, M.; Falcidieno, B.; Spagnuolo, M.: Hierarchical mesh segmentation based on fitting primitives. The Visual Computer, 22(3), 181–193, 2006. http://doi.org/10.1007/s00371-006-0375-x.

[3] Benhabiles, H.; Lavoué, G.; Vandeborre, J.P.; Daoudi, M.: Learning boundary edges for 3D-mesh segmentation. In Computer Graphics Forum, vol. 30, 2170–2182. Wiley Online Library, 2011. http://doi.org/10.1111/j.1467-8659.2011.01967.x.

[4] Bergamasco, F.; Albarelli, A.; Torsello, A.: A graph-based technique for semi-supervised segmentation of 3D surfaces. Pattern Recognition Letters, 33(15), 2057–2064, 2012. http://doi.org/10.1016/j.patrec.2012.03.015.

[5] Boulch, A.; Le Saux, B.; Audebert, N.: Unstructured point cloud semantic labeling using deep segmentation networks. 3DOR, 2, 7, 2017. http://doi.org/10.2312/3dor.20171047.

[6] Cai, N.; Anwer, N.; Scott, P.J.; Qiao, L.; Jiang, X.: A new partitioning process for geometrical product specifications and verification. Precision Engineering, 62, 282–295, 2020. http://doi.org/10.1016/j.precisioneng.2019.12.009.

[7] Chiabert, P.; Costa, M.: Statistical modelling of nominal and measured mechanical surfaces. J. Comput. Inf. Sci. Eng., 3(1), 87–94, 2003. http://doi.org/10.1115/1.1569941.

[8] Cohen-Steiner, D.; Alliez, P.; Desbrun, M.: Variational shape approximation. In ACM SIGGRAPH 2004 Papers, 905–914, 2004. http://doi.org/10.1145/1186562.1015817.

[9] Friedrich, M.; Illium, S.; Fayolle, P.A.; Linnhoff-Popien, C.: A hybrid approach for segmenting and fitting solid primitives to 3D point clouds. In VISIGRAPP (1: GRAPP), 38–48, 2020. http://doi.org/10.5220/0008870600380048.

[10] Gelfand, N.; Guibas, L.J.: Shape segmentation using local slippage analysis. In Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing, 214–223, 2004. http://doi.org/10.1145/1057432.1057461.

[11] Guerry, J.; Boulch, A.; Le Saux, B.; Moras, J.; Plyer, A.; Filliat, D.: Snapnet-r: Consistent 3D multi-view semantic labeling for robotics. In Proceedings of the IEEE International Conference on Computer Vision Workshops, 669–678, 2017. http://doi.org/10.1109/ICCVW.2017.85.

[12] Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; Bennamoun, M.: Deep learning for 3D point clouds: A survey. IEEE transactions on pattern analysis and machine intelligence, 43(12), 4338–4364, 2020. http://doi.org/10.1109/TPAMI.2020.3005434.

[13] ISO: Geometrical product specifications (gps) general concepts part 1: Model for geometrical specifications and verification. ISO 17450-1:2011, 2011.

[14] Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A.: Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, 1125–1134, 2017. http://doi.org/10.1109/CVPR.2017.632.

[15] Koch, S.; Matveev, A.; Jiang, Z.; Williams, F.; Artemov, A.; Burnaev, E.; Alexa, M.; Zorin, D.; Panozzo, D.: Abc: A big cad model dataset for geometric deep learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9601–9611, 2019.

[16] Komarichev, A.; Zhong, Z.; Hua, J.: A-CNN: Annularly convolutional neural networks on point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7421–7430, 2019. http://doi.org/10.1109/CVPR.2019.00760.

[17] Lai, Y.K.; Hu, S.M.; Martin, R.R.; Rosin, P.L.: Fast mesh segmentation using random walks. In

Proceedings of the 2008 ACM symposium on Solid and physical modeling, 183–191, 2008. http://doi.org/10.1145/1364901.1364927.

[18] Lavoué, G.; Dupont, F.; Baskurt, A.: A new CAD mesh segmentation method, based on curvature tensor analysis. Computer-Aided Design, 37(10), 975–987, 2005. http://doi.org/10.1016/j.cad.2004.09.001.

[19] Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; Chen, B.: PointCNN: Convolution on x-transformed points. Advances in neural information processing systems, 31, 820–830, 2018. http://doi.org/10.48550/arXiv.1801.07791.

[20] Maturana, D.; Scherer, S.: Voxnet: A 3D convolutional neural network for real-time object recognition. In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 922–928. IEEE, 2015. http://doi.org/10.1109/ACCESS.2019.2919332.

[21] Pauly, M.; Gross, M.; Kobbelt, L.P.: Efficient simplification of point-sampled surfaces. In IEEE Visualization, 2002. VIS 2002., 163–170. IEEE, 2002. http://doi.org/10.1109/VISUAL.2002.1183771.

[22] Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J.: Pointnet: Deep learning on point sets for 3D classification and segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, 652–660, 2017. http://doi.org/10.1109/CVPR.2017.16.

[23] Qie, Y.; Anwer, N.: Toward non-default partitioning for compound feature identification in engineering design. Procedia CIRP, 100, 852–857, 2021.

[24] Qie, Y.; Anwer, N.; Scott, P.J.; Jiang, J.X.; Srinivasan, V.: Toward a mathematical definition of reconstruction operation for iso gps standards. Procedia CIRP, 92, 152–157, 2020. http://doi.org/10.1016/j.procir.2020.05.183.

[25] Qie, Y.; Qiao, L.; Anwer, N.: Enhanced invariance class partitioning using discrete curvatures and conformal geometry. Computer-Aided Design, 133, 102985, 2021. http://doi.org/10.1016/j.cad.2020.102985.

[26] Rabbani, T.; Van Den Heuvel, F.; Vosselmann, G.: Segmentation of point clouds using smoothness constraint. International archives of photogrammetry, remote sensing and spatial information sciences, 36(5), 248–253, 2006.

[27] Raina, P.; Mudur, S.; Popa, T.: Sharpness fields in point clouds using deep learning. Computers & Graphics, 78, 37–53, 2019. http://doi.org/10.1016/j.cag.2018.11.003.

[28] Rodrigues, R.S.; Morgado, J.F.; Gomes, A.J.: Part-based mesh segmentation: a survey. In Computer Graphics Forum, vol. 37, 235–274. Wiley Online Library, 2018. http://doi.org/10.1111/cgf.13323.

[29] Romanengo, C.; Raffo, A.; Qie, Y.; Anwer, N.; Falcidieno, B.: Fit4CAD: A point cloud benchmark for fitting simple geometric primitives in cad objects. Computers & Graphics, 102, 133–143, 2022. http://doi.org/10.1016/j.cag.2021.09.013.

[30] Schleich, B.; Anwer, N.; Mathieu, L.; Wartzack, S.: Skin model shapes: A new paradigm shift for geometric variations modelling in mechanical engineering. Computer-Aided Design, 50, 1–15, 2014. http://doi.org/10.1016/j.cad.2014.01.001.

[31] Schnabel, R.; Wahl, R.; Klein, R.: Efficient ransac for point-cloud shape detection. In Computer graphics forum, vol. 26, 214–226. Wiley Online Library, 2007. http://doi.org/10.1111/j.1467-8659.2007.01016.x.

[32] Shamir, A.: A survey on mesh segmentation techniques. In Computer graphics forum, vol. 27, 1539–1556. Wiley Online Library, 2008. http://doi.org/10.1111/j.1467-8659.2007.01103.x.

[33] Su, H.; Maji, S.; Kalogerakis, E.; Learned-Miller, E.: Multi-view convolutional neural networks for 3D shape recognition. In Proceedings of the IEEE international conference on computer vision, 945–953, 2015. http://doi.org/10.1109/ICCV.2015.114.

[34] Tchapmi, L.; Choy, C.; Armeni, I.; Gwak, J.; Savarese, S.: Segcloud: Semantic segmentation of 3D point clouds. In 2017 international conference on 3D vision (3DV), 537–547. IEEE, 2017. http://doi.org/10.1109/3DV.2017.00067.

[35] Theologou, P.; Pratikakis, I.; Theoharis, T.: A comprehensive overview of methodologies and performance evaluation frameworks in 3D mesh segmentation. Computer Vision and Image Understanding, 135, 49–82, 2015. http://doi.org/10.1016/j.cviu.2014.12.008.

[36] Vosselman, G.; Gorte, B.G.; Sithole, G.; Rabbani, T.: Recognising structure in laser scanner point clouds. International archives of photogrammetry, remote sensing and spatial information sciences, 46(8), 33–38, 2004.

[37] Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M.: Dynamic graph CNN for learning on point clouds. ACM Transactions On Graphics (tog), 38(5), 1–12, 2019. http://doi.org/10.1145/3326362.

[38] Williams, R.M.; Ilieș, H.T.: Practical shape analysis and segmentation methods for point cloud models. Computer Aided Geometric Design, 67, 97–120, 2018. http://doi.org/10.1016/j.cagd.2018.10.003.

[39] Xiao, D.; Lin, H.; Xian, C.; Gao, S.: CAD mesh model segmentation by clustering. Computers & Graphics, 35(3), 685–691, 2011. http://doi.org/10.1016/j.cag.2011.03.020.

[40] Yan, S.; Yang, Z.; Ma, C.; Huang, H.; Vouga, E.; Huang, Q.: HPNet: Deep primitive segmentation using hybrid representations. arXiv preprint arXiv:2105.10620, 2021. http://doi.org/10.48550/arXiv.2105.10620.

[41] Zhang, H.; Van Kaick, O.; Dyer, R.: Spectral mesh processing. In Computer graphics forum, vol. 29, 1865–1894. Wiley Online Library, 2010. http://doi.org/10.1111/j.1467-8659.2010.01655.x.

[42] Zhang, J.; Zhao, X.; Chen, Z.; Lu, Z.: A review of deep learning-based semantic segmentation for point cloud (november 2019). IEEE Access, 2019. http://doi.org/10.1109/ACCESS.2019.2958671.

[43] Zhang, J.; Zheng, J.; Cai, J.: Interactive mesh cutting using constrained random walks. IEEE Transactions on Visualization and Computer Graphics, 17(3), 357–367, 2010. http://doi.org/10.1109/TVCG.2010.57.

[44] Zhu, Z.; Anwer, N.; Huang, Q.; Mathieu, L.: Machine learning in tolerancing for additive manufacturing. CIRP Annals, 67(1), 157–160, 2018.