XAI FOR POINT CLOUD DATA USING PERTURBATIONS BASED ON MEANINGFUL SEGMENTATION

RAJU NINGAPPA MULAWADE
ZFT & UX-Vis
Hochschule Worms University of Applied Sciences
Erenburgerstr. 19
Worms, 67549, Germany
mulawade@hs-worms.de

CHRISTOPH GARTH
Scientific Visualization Lab
RPTU Kaiserslautern-Landau
Gottlieb-Daimler-Str.
Kaiserslautern, 67663, Germany
garth@rptu.de

ALEXANDER WIEBEL
ZFT & UX-Vis
Hochschule Worms University of Applied Sciences
Erenburgerstr. 19
Worms, 67549, Germany
wiebel@hs-worms.de

August 2025

ABSTRACT

In this work, we propose a novel segmentation-based explainable artificial intelligence (XAI) method for neural networks working on point cloud classification. As one building block of this method, we also propose a novel point-shifting mechanism to introduce perturbations in point cloud data.

In the last decade, Artificial intelligence (AI) has seen an exponential growth. However, due to the "black-box" nature of many of these AI algorithms, it is important to understand their decision-making process when it comes to their application in critical areas. Our work focuses on explaining AI algorithms that classify point cloud data. An important aspect of the methods used for explaining AI algorithms is their ability to produce explanations that are easy for humans to understand. This allows the users to analyze the performance of AI algorithms better and make appropriate decisions based on that analysis. Therefore, in this work, we intend to generate meaningful explanations that can be easily interpreted by humans. The point cloud data considered in this work represents 3D objects such as cars, guitars, and laptops. We make use of point cloud segmentation models to generate explanations for the working of classification models. The segments are used to introduce perturbations into the input point cloud data and generate saliency maps. The perturbations are introduced using the novel point-shifting mechanism proposed in this work which ensures that the shifted points no longer influence the output of the classification algorithm.

In contrast to any previous methods, the segments used by our method are meaningful, i.e. humans can easily interpret the meaning of these segments. Thus, the benefit of our method over other methods is its ability to produce more meaningful saliency maps. We compare our method with the use of classical clustering algorithms to generate explanations. We also analyze the saliency maps generated for some example inputs using our method to demonstrate the usefulness of our proposed method in generating meaningful explanations.

***K*eywords** Artificial intelligence · explainable AI · point cloud data · segmentation

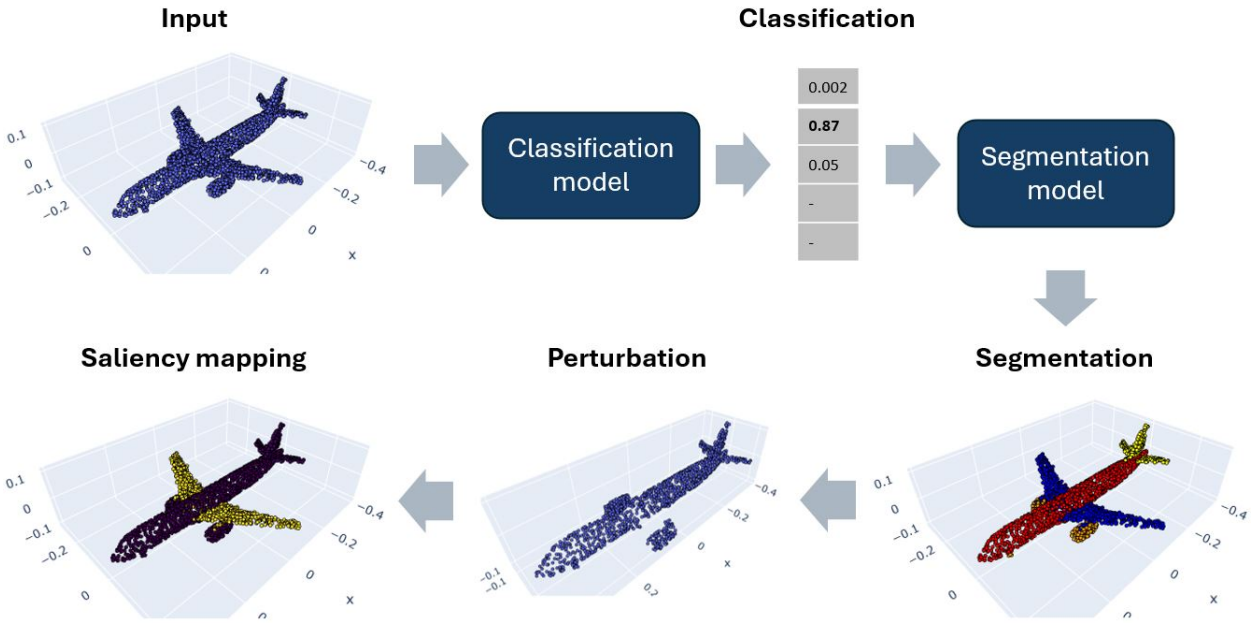


Figure 1: An overview of the XAI pipeline proposed.

1 Introduction

Explainable artificial intelligence (XAI) has become an important field of research in the last decade. This is mainly due to the exponential growth in AI which is finding use in almost every field of application from agriculture to autonomous vehicles. AI algorithms are now capable of performing various difficult tasks with high accuracy. This has prompted industries belonging to various fields to incorporate AI algorithms and improve the performance of various tasks performed in those industries. However, most AI algorithms performing challenging tasks have complex architecture which makes it highly difficult to understand how the algorithm is making a decision. Therefore, many such AI models are referred to as "black boxes". This is one of the primary concerns related to AI that hinders the use of AI algorithms in high-risk tasks. Thus, as AI algorithms learn to perform more complex tasks, the need to understand their decision-making process becomes more important. Our work contributes to this important field of research.

AI algorithms work on various types of data such as text, tabular, image, and point clouds. In this work, we propose an explainability method that focuses on explaining the classification models working on point cloud data as we see a growing trend in the use of point cloud data for AI model development in the last decade [5] [2]. We also observe a similar trend in XAI research work targeting algorithms working on point cloud data [11] [27] [9] [28]. However, there is still a significant gap between the XAI work developed for data types such as image and text compared to point cloud data. Therefore, through our work, we attempt to reduce this gap by contributing a method based on meaningful segmentation to the point cloud-based XAI field of research. Figure 1 shows an overview of our proposed method.

The main contributions of our work are:

- Segmentation-based XAI for understanding classification networks working on point cloud data. The proposed method is a perturbation-based method and it is model-agnostic.
- Proposal of a novel point shifting mechanism for the perturbation of point cloud data.
- Two types of introducing perturbations to generate explanations that provide different interesting insights.
- Detailed analysis of the proposed method against clustering-based methods to highlight its advantages. The analysis shows how the proposed method generates meaningful explanations.

The rest of the paper is organized as follows: Section 2 gives a detailed overview of the literature that is relevant to our work. Section 3 describes the proposed XAI method in this work and the perturbation mechanisms utilized for generating saliency maps. It also provides an overview of the data and AI models used in this work. Section 4 provides a detailed analysis of the proposed method using multiple examples to indicate the usefulness of our method. Section 5 contains our final remarks regarding the work and the direction in which the future work can progress.

2 Related Work

As point cloud data is gaining importance in AI developments, the research related to explaining AI algorithms working on point cloud data has also seen an upward trajectory. Many authors have attempted to provide explanations for these algorithms employing various types of explainability mechanisms. Mulawade et al. [11] have provided a detailed survey of all the XAI literature addressing the issue of explainability for AI models working on point cloud data. Saranti et al. [14] provided a survey focusing specifically on the explainability of graph neural networks (GNNs) working on point cloud data. Among the different types of XAI methodologies proposed in the past, the perturbation-based methods have found greater importance in explaining point cloud-based AI models. This is evident in the list of papers surveyed by the authors in [11] with papers proposing perturbation-based XAI methods being the highest in number among the methodologies used. Some of the most prominent perturbation-based methods (considering all types of data such as image, text, and point clouds) are SHapley Additive exPlanations (SHAP) [7] and Local Interpretable Model-agnostic Explanations (LIME) [13]. The perturbation-based XAI methods for point cloud data use different types of perturbation to generate explanations for the working of AI models. We describe them below and highlight the need for our work.

Zheng et al.[28] proposed an XAI method that computes saliency maps by introducing perturbation into the input data. The perturbation method used in this work uses the process of moving a specific point to the center of the point clouds to introduce perturbations in the input data. The authors consider the spherical coordinate system to compute the attributions corresponding to the points as they are gradually shifted to the center of the input data.

Taghanaki et al.[19] proposed a perturbation-based XAI method for explaining classification networks working on point cloud data. They proposed a method called *PointMask* which learns to mask out points in the input data based on their contribution to the output class score.

Shen et al.[16] proposed a perturbation-based XAI method for analyzing the classification network working on point cloud data. The authors used Shapley values[15] to compute saliency maps. The input point cloud is segmented into a fixed number of regions and points belonging to specific regions are moved to the center of the point cloud data to measure the changes in the output target class to generate a saliency map.

Verbung [25] proposed a perturbation-based XAI method for understanding a segmentation model working on point cloud data. The author introduced perturbations into the input data by modifying specific regions (such as the shape of a manually selected part of an object) in the input point cloud data and measuring the effect of this perturbation on the segmentation output.

Tan et al.[23] proposed an XAI method that adapts LIME [13] to explain the decision-making process of classification models working on point clouds. The point cloud data is divided into multiple regions using a clustering method and perturbations are introduced using these clusters to compute saliency maps using the LIME methodology.

Tan [21] proposed another perturbation-based XAI method for point cloud-based AI models that perform a classification task. In this method, the target output class score is maximized by modifying manually selected parts of the input data. The authors made use of autoencoders to encode and generate new input samples.

Tan [20] also proposed an XAI method for understanding a classification network working on point cloud data that is based on feature ablation. The author proposed removing specific features (identified by the author) from all the data instances in the training dataset and retraining the model on the perturbed data. The change in classification accuracy achieved by the model is then used as an attribution that indicates the importance of the removed features.

Tan and Kotthaus[24] used integrated gradients[18] to identify critical points in the input point cloud data and use these critical points to perturb the input data.

Miao et al.[10] proposed Learnable Randomness Injection (LRI) that provides an explanation for the working of a classification model with point cloud data as its input. The proposed method learns to inject randomness (perturbation) into the input data during the training process taking into consideration the performance of the AI model in classifying the data.

The most recent contribution of Tan[22] to the topic proposes an activation-flow-based AM method named Flow AM that makes use of the activation maximization of the output target class and also forces the neurons in the intermediate layers to align their activation values to the values that correspond to actual input instances during this process.

Atik et al.[1] adapted SHAP for interpretation of the classification model working on photogrammetric point cloud data. The authors mainly focused on the explainability of ensemble classifiers in this work.

Another adaptation of Shapley values for understanding point cloud-based AI models was proposed by Shen et al.[17] where the authors divided the input point cloud data uniformly into multiple regions and computed Shapley values. The

perturbation method used in this work was the "point shifting" mechanism where the points of some regions are moved to the center of the point cloud data.

Lavasa et al.[6] adapted SHAP for analyzing the performance of AI models that predict the accuracy of laser scanning devices.

However, none of the above methods mention or describe using *meaningful* segments to introduce perturbations into the input data unless introduced manually by the developers of the methods. The use of meaningless segments for perturbation leads to the generation of saliency maps with attributions assigned to data points that are difficult to interpret. Furthermore, methods employing the perturbation mechanism by shifting or removing individual points are computationally expensive. In addition to this, we also believe that individual points do not carry important information such as structural information that is crucial information in point clouds. This important information is captured by a set of points. Therefore, the perturbation mechanism should consider using sets of multiple points in the point cloud data to introduce perturbations instead of individual points. Furthermore, the information captured by these sets of points should be meaningful. This leads to the generation of saliency maps that are meaningful, and therefore, easy for humans to interpret. In addition to this, perturbations introduced into the input data should generate an input where specific features have no influence on the output.

In this work, we propose a perturbation-based method that makes use of meaningful segments generated by an algorithm to perturb the input data and compute attributions based on the change in the output value of the target class. We also propose a point-shifting mechanism for introducing perturbations in point cloud data that meets the requirement mentioned above.

3 Segmentation-based XAI

The proposed segmentation-based XAI method for point cloud classification models utilizes a segmentation model that generates meaningful segments from the given input point cloud data. An overview of the proposed method is shown in Figure 1. It consists of four stages:

- Classification
- Segmentation
- Perturbation
- Saliency mapping

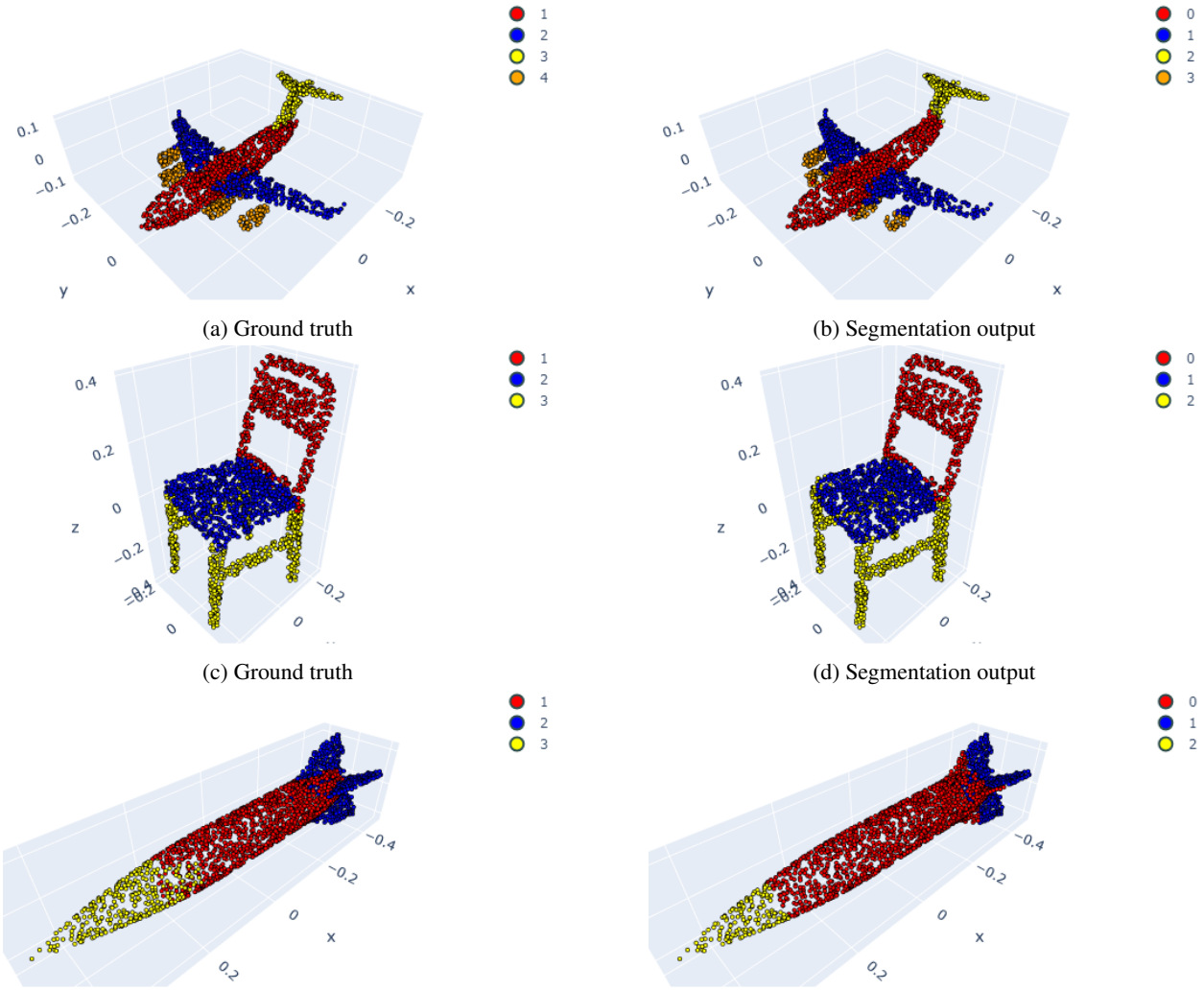
In the first stage, the input point cloud data is used as the input for the classification model which predicts the output class of this data. This is the same classification model that we intend to understand in the XAI process. Based on the output class, the corresponding segmentation model is chosen from the list of pre-trained models. In the second stage, the selected segmentation model is used to meaningfully segment the input point cloud data. The resulting segments are used in the third stage to perturb the input data. Using the classification model and the perturbed input data, a saliency map is computed in the final stage of this pipeline.

3.1 Segmentation

In our XAI method, we intend to segment the input point cloud data in a meaningful way. This means that the segments produced by the segmentation mechanism are easy to understand for humans. For example, the segmentation of point cloud data representing a human 3D model into segments that represent the head, hands, legs, and torso. In our work, we use two segmentation mechanisms to divide the input point cloud data into multiple meaningful segments. These are explained below.

3.1.1 Segmentation mechanism

This mechanism consists of AI models that are trained for part segmentation tasks on the point cloud data. These models use the same input data that is used by the classification model, identify different meaningful segments in the data, and assign them with specific labels. The dataset used in our work consists of point cloud data instances representing 16 types of 3D models. To ensure better performance, we train segmentation models to segment data instances representing a specific 3D model such as airplanes or cars instead of training one single segmentation model to segment point clouds representing every kind of 3D model. Therefore, we have 16 segmentation models with each model catering to segmenting a specific type of point cloud data. Figure 2 shows the segmentation of point clouds representing an airplane, a chair, and a rocket by the three corresponding segmentation models.



(a) Ground truth

(b) Segmentation output

(c) Ground truth

(d) Segmentation output

(e) Ground truth

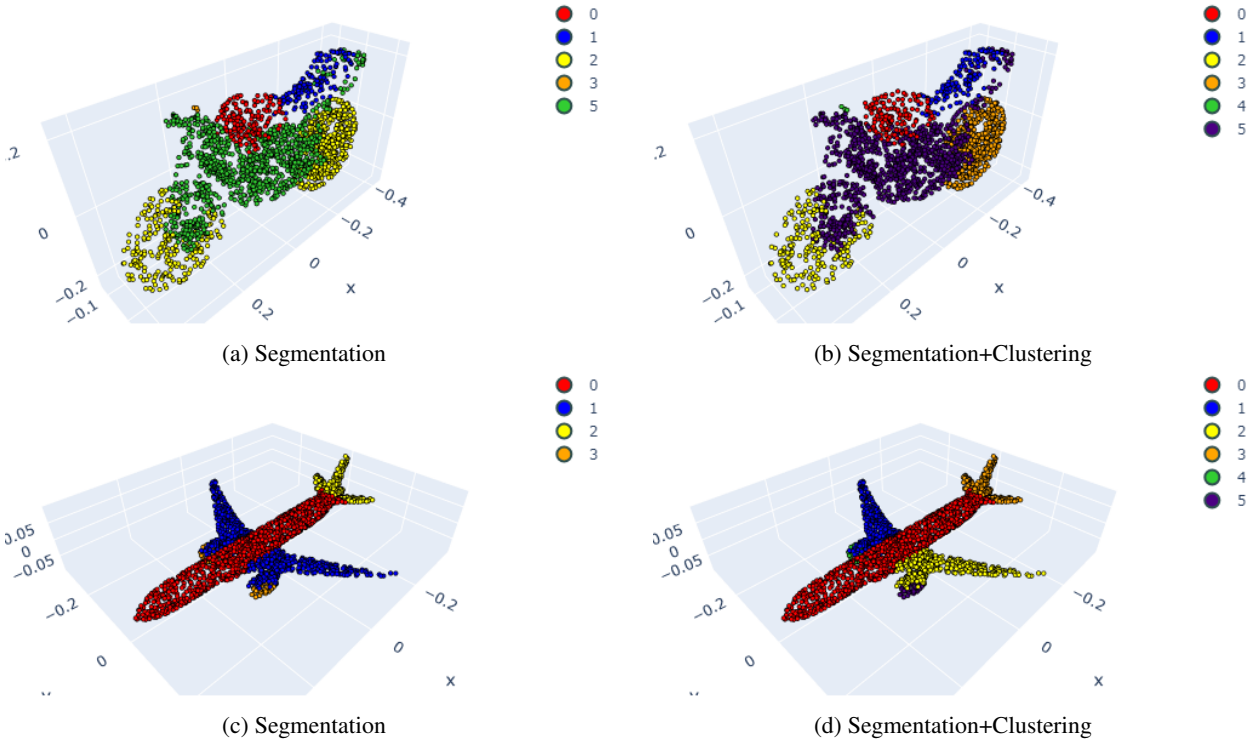
(f) Segmentation output

Figure 2: Examples of point clouds segmentation performed by our segmentation models and their corresponding ground truth labels.

3.1.2 Segmentation+Clustering mechanism

The dataset taken into consideration in this work contains point clouds representing various types of 3D models. Some of these models contain features that consist of more than one part, such as the four wheels of cars, the two wheels of motorbikes and the two wings of airplanes. The segmentation algorithms classify these features as one class/group. This leads to perturbations where all parts of these features are shifted (in case of *presence of a feature* mechanism, see 3.2.2) to a chosen point or retained (in case of *absence of a feature* mechanism, see 3.2.1) in the input with remaining segments shifted to the chosen point. This leads to the generation of saliency maps that contain the same saliency attribution value attached to these features belonging to a single class. This can be observed for the wings of airplanes in the saliency maps visualized in Figure 4c and Figure 6a and their corresponding bar plots Figure 4d and Figure 6b. However, the relevance of these multiple features that are grouped into one class is not identical in many cases. Therefore, it can be important to understand the influence these individual features have on the output class score in addition to their influence as a group.

To generate saliency maps for individual features, we made use of clustering algorithms for the segmentation-based mechanism to further segment the input point cloud data. The segmentation model's output is used by the clustering algorithms to further cluster the data. We use two clustering algorithms: 1) DBSCAN [4] for determining the number of clusters in a given segment of the segmentation output. 2) KMeans [8] clustering algorithm to cluster the given



(a) Segmentation

(b) Segmentation+Clustering

(c) Segmentation

(d) Segmentation+Clustering

Figure 3: Clustering of the segments produced by the segmentation models to obtain finer segments.

segment based on the number of clusters determined by DBSCAN. We use this combination because the DBSCAN algorithm tends to classify some of the outlying points of the segment as outliers and assigns a separate value to them. This leads to undesired clusters. To avoid this, we combine it with KMeans which takes the number of clusters (without taking outliers' class into account) as input and produces the desired number of clusters. Figure 3 shows examples of the segmentation of point clouds representing a motorbike and an airplane using the segmentation and segmentation+clustering methods. Figure 3a is the output of the segmentation model that identifies the wheels of the motorbike as one segment. Similarly, the wings and engines are labeled as a single segment each, as shown in Figure 3c. The segmentation+clustering algorithm clusters the wheels of the motorbike to produce front and rear wheels as shown in Figure 3b. The method also clusters the wings of the airplane into two separate clusters. A similar result is observed with respect to the engines of the airplane as shown in Figure 3d.

3.2 Perturbation mechanism

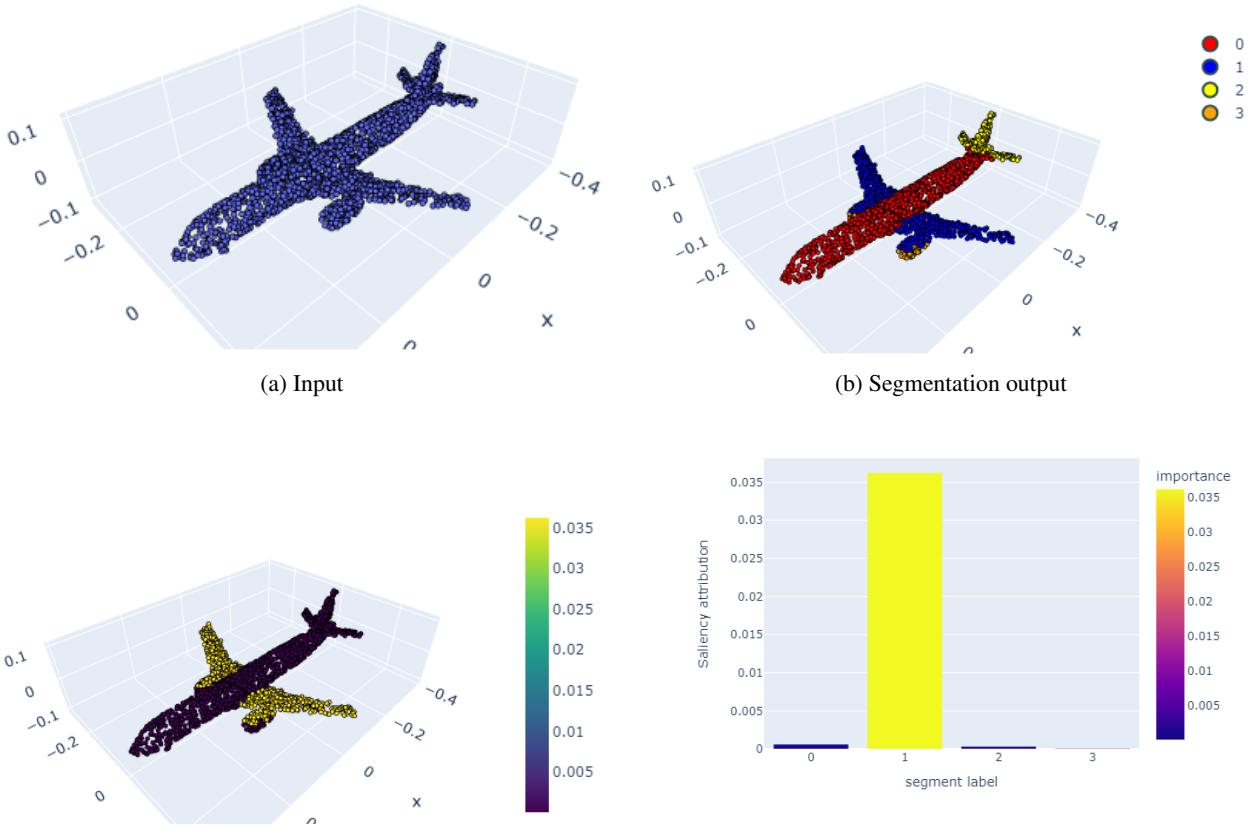
As mentioned in section 1, the proposed XAI method in this work is a perturbation-based method. We use two types of perturbation introducing mechanisms to generate saliency attributions providing interesting insights into the working of the classification model. We explain these mechanisms and the rationale behind them below.

3.2.1 Absence of a Feature

The perturbation mechanism used in this work introduces perturbation by *removing* a specific segment from the input data. Removing here refers to shifting all the points belonging to this specific segment to a chosen point in the input data. A segment can be an individual feature (e.g. bonnet in a car) or a collection of similar features (e.g. wheels of a car). The perturbed data instance is then used as input for the classification model to compute saliency attributions. The saliency attributions are computed as follows:

$$S_{AF}(x) = |P(a) - P(a')| \tag{1}$$

where $S_{AF}(x)$ is the saliency attribution corresponding to the segment x that is used for perturbation, a is the actual input, a' is the perturbed input, and $P(s)$ refers to the output class score by the classification model for a given input s. Figure 4 shows an example of this saliency mapping method for input point cloud data representing an airplane.



(a) Input

(b) Segmentation output

(c) Saliency map

(d) Saliency attributions

Figure 4: Saliency map produced by our method for the given input point cloud data representing a plane. Note: Refer to the 'segmentation output' figure (b) for corresponding parts represented along the x-axis in the bar plot (d).

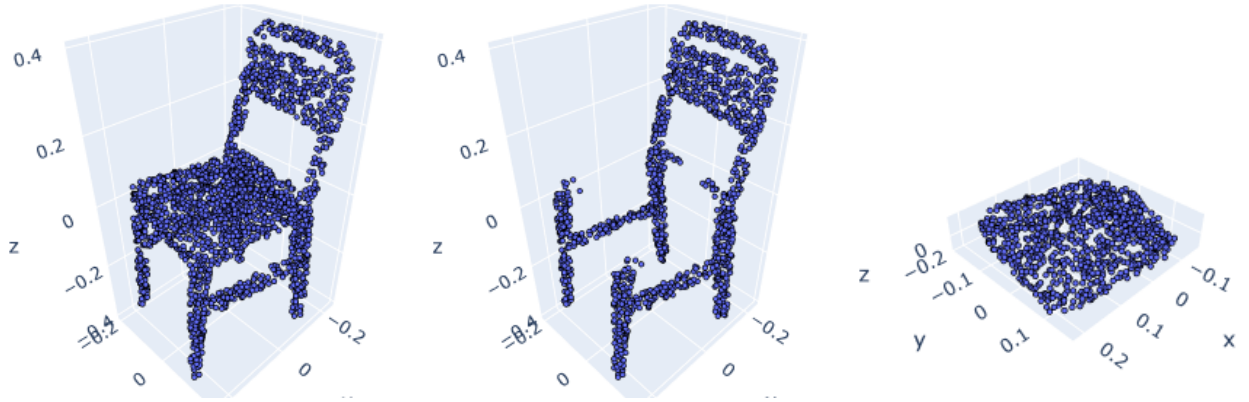
3.2.2 Presence of a Feature

In addition to the above-mentioned method, we propose a variation of it where we analyze the impact of individual features (or segments) on the output data. Here, we introduce perturbation into the input data by *retaining* a specific segment and moving all the points belonging to other segments to the center of the point cloud data. Mathematically, it can be expressed as follows:

$$S_{PF}(x) = -|(P(a) - P(a"))| \tag{2}$$

where $S_{PF}(x)$ is the saliency attribution corresponding to the segment x, a is the actual input, a" is the perturbed input where the points not belonging to the segment x are moved to the chosen point, and $P(s)$ refers to the output class score by the classification model for a given input s. The minus sign $(-)$ is used for the visualization purpose. It allows the segment having the highest influence on the output value to have the highest attribution while the lowest influential segment has the lowest value.

The saliency attributions generated by this method can be interpreted as a measure of the influence an individual segment has on the output prediction value when it is the only segment present in the input. This informs us about how good a specific segment is in carrying crucial information on its own. This interpretation is slightly different from the previous one (described in 3.2.1) as it does not provide the model with any other information that is carried by other segments or the information that is generated when we combine two or more segments as shown in Figure 5 where the perturbed data (Figure 5b) manages to capture the structure of chair even after the points belonging to one segment (Figure 5c) are moved to the center of the data. Therefore, we decided to look into how much information a single segment carries that is independent of all the other segments. Figure 6 shows the saliency attributions computed using this perturbation mechanism for the same input point cloud as data considered in Figure 4.



(a) Input data (b) Seat part moved (c) Seat part

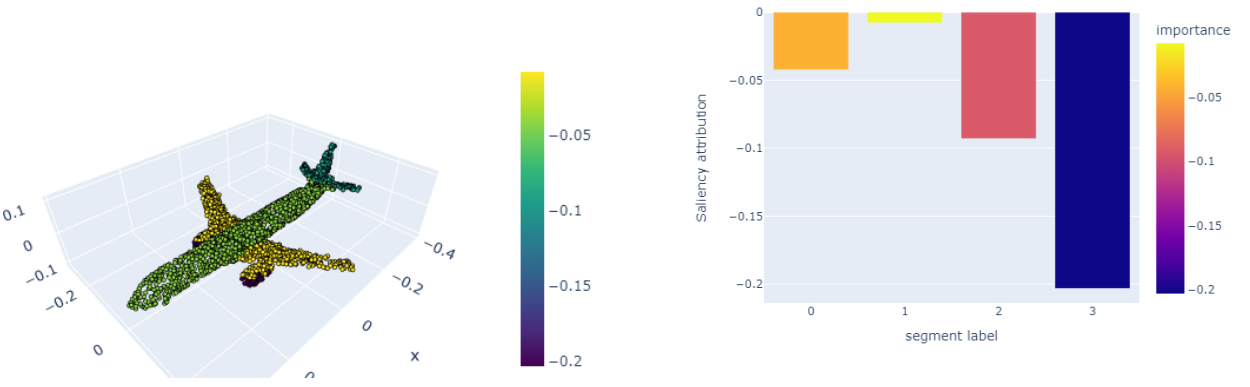
Figure 5: Structural information carried by a point cloud data (left figure) and its perturbed data where all the points belonging to one segment (right figure) are moved to the center of the data (center figure).

3.3 Data and Model

The dataset used in this work contains the part segmentation of a subset of ShapeNetCore[3] models based on the work of Yi et al.[26]. The dataset consists of ∼16000 models from 16 shape categories and the number of data instances in each category varies from 55 to 5266. The number of parts for each model in each category also varies from two to six as each category consists of different types of 3D models representing a specific object such as an airplane. We use this dataset for both classification and segmentation tasks.

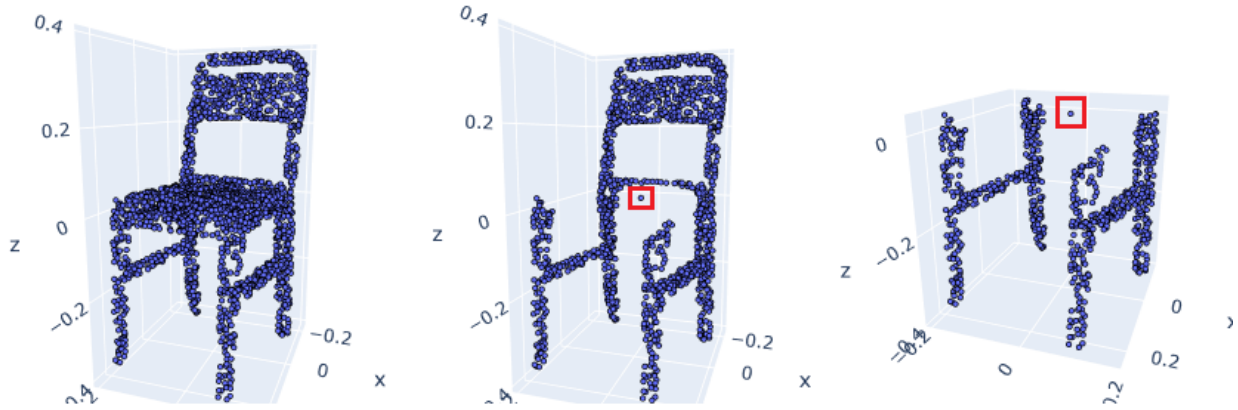
We use classification and segmentation networks based on the PointNet[12] architecture. The classification model is trained on the above-mentioned dataset containing point clouds of 16 different categories. We trained two classification models, one with the default orientation of the point cloud objects in the dataset and the other with the augmented dataset where we modify the orientation of the point cloud objects. We trained the former classification model (that uses the data with default orientation) for 10 epochs with the stochastic gradient descent (SGD) optimizer at 0.001 learning rate and the latter (with augmented data) for 100 epochs (because of the increased complication in the input data due to the augmentation) while keeping the remaining hyperparameters unchanged.

To obtain better segmentation results, we trained individual models to segment particular point cloud data types. Our dataset consists of point clouds representing 16 types of 3D objects such as airplanes, tables, and cars. Thus, we trained 16 segmentation models with each model focusing on segmenting the point cloud data representing a specific 3D object. We also augmented the training data for these segmentation models by modifying the orientation of the data instances.



(a) Saliency map (b) Saliency attributions

Figure 6: Saliency map produced by using the "presence of feature" method for the input used in Figure 4. Note: Refer to the 'segmentation output' figure (b) in Figure 4 for corresponding parts represented along the x-axis in the bar plot.



(a) Input point cloud (b) Perturbed input (c) Perturbed input

Figure 7: Shifting points to the centroid of the input point cloud data. Points representing the seat (in (b)) and seat & backrest (in (c)) are shifted to the center of the input data (marked with red rectangle).

This is similar to the augmentation performed on the training data for classification models. We trained these models with SGD optimizer at 0.001 learning rate with the number of epochs varying from 20 to 150 depending on the size of the subsets representing the 3D objects. Figure 2 shows some examples that demonstrate the performance of our trained segmentation models on their corresponding input data.

The output of these segmentation models will be used to introduce perturbations into input data instances as described above. We use the point-shifting mechanism to introduce perturbations as it allows the input point cloud instance to retain its number of points thereby avoiding complications in the saliency method. The point-shifting mechanism is described in the following subsection.

3.4 Point Shifting Mechanism

To introduce perturbations into the input data, as mentioned in subsection 3.3, we use the point-shifting mechanism. Zheng et al.[28] proposed the idea of shifting the points to the center of the input data instead of dropping them from the input. This is based on the intuition that all the outward points in the point cloud determine the output class score of the classification model as they encode shape information while the points closer to the center of the point cloud have almost no influence. However, this process of shifting the points to the center of the point clouds does not fit well with our work. For example, Figure 7 shows an example of the perturbation of the input data by moving the points belonging to two segments (seat and backrest of the chair) to the center (marked by a red rectangle) of the input data. Thus, the center of the input data now contains a large number of input points and is not actually a part of the retained structure. Therefore, it can act as an additional feature in the input data which is undesirable as we expect the shifted points to have no influence on the decision-making process.

To address this issue, we need to determine a point in the input space where the points belonging to the specific segment can be shifted, and the shifted points do not influence the output class score. This is possible when the shifted points do not add any structural information to the data. Shifting the points to the center of the retained structure does not always fulfill this requirement. This is evident in Figure 7c where the center of the retained structure (the legs of the chair) lies in between the leg structures and thus acts as an additional structure in the perturbed data.

One feasible solution is when the selected point for shifting the points is itself a part of the retained structure in the perturbed input data. This will allow the shifted points to be a part of the perturbed data and provide no additional structural information for the classification model. Since we have multiple points in the retained structure in the input data, we choose a random point from it for shifting the points to. We observe that the saliency attributions corresponding to the features do not vary when selecting random points for perturbation. We discuss this mechanism with some examples in section 4.

4 Results and Discussion

In this section, we evaluate our method using various examples and criteria to highlight the usefulness of the mechanisms that are part of our proposed method.

4.1 Clustering-based method

In this subsection, we analyze the use of classical clustering algorithms for segmenting point cloud data, indicate the issues associated with their use, and describe how our method overcomes these issues. For the analysis, we used clustering algorithms such as the k-means algorithm to generate clusters in the input point cloud data and use these clusters to perturb the same input data to compute saliency attributions. Figure 8 shows examples of using the KMeans clustering method with varying numbers of clusters, c, for generating segments for computing the saliency maps. We used the *absence of feature* mechanism to introduce perturbations and compute saliency attributions. We observed that the saliency maps differ as we vary the number of clusters. The most important part for $c = 3$ is the top part of the chair, and as we increase c to 12, the most important part shifts to the front left corner of the seat and the front bottom part of the right leg of the chair. We observed similar behavior with other clustering methods such as *spectral* and *agglomerative clustering*. This makes the use of clustering methods for XAI methods tailored for point cloud data unreliable. Our proposed method addresses this issue by using segmentation models that are trained to segment a given point cloud data into a specific number of segments. Another major advantage of using segmentation models over classical clustering algorithms is their ability to learn and adapt to new types of data instances. In other words, we can improve the performance of segmentation models by training it on more data whereas the classical clustering algorithms do not offer this flexibility.

4.2 Use of Random point

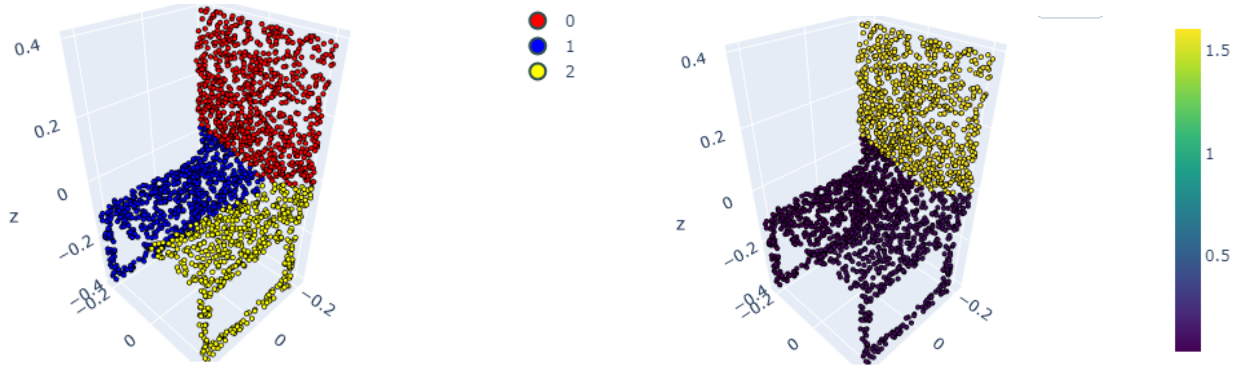
As mentioned in subsection 3.4, it is important to find a point in the input space where the points belonging to selected segments can be shifted, and these shifted points do not add any structural information to the perturbed data. In this section, we use an example to discuss and understand how our proposed method of selecting a random point in the retained structure yields better results than other methods such as moving the points to the origin or to the center of the point cloud data.

Figure 9 shows examples of input data perturbation for an input data representing an airplane. Figure 9b is the segmentation output obtained from a segmentation model which is used to introduce perturbations into the input data shown in Figure 9a. For this example, we selected the segment representing the wings to introduce perturbations. We selected a random point in the retained structure (structure without wings) and shifted the points belonging to the segment representing wings. Two examples are shown in Figure 9 with one random point selected in the tail region of the airplane (see Figure 9c) while the other random point is selected in the central part of the fuselage (see Figure 9d). We shift the points representing the wings to these random points and use these two perturbed data instances to analyze the effect of the perturbations. We use them as input for the classification model and study the change in the output values. We observed that the output values (all 16 values in the output vector) did not change. In other words, the choice of point in the retained structure had no influence on the output values. We observed a similar pattern when we chose different points in the retained structure to shift the points. This observation strengthened our intuition that when the shifted points are a part of the retained structure (irrespective of the point selected in the retained structure for the shifting process), they do not provide any additional structural information for the classification model. Therefore, we use the random point selection mechanism for our point-shifting process.

4.3 Effects of Different Feature Instances: Wings vs. Fuselage

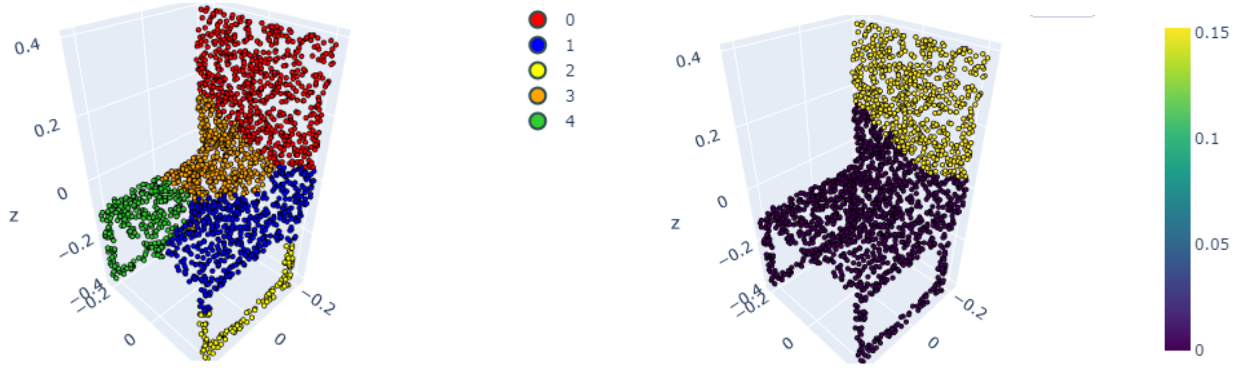
The datasets used for training classification models usually contain a large number of samples representing different classes. In addition to the differences between the samples representing each class, samples representing a specific class also vary slightly with respect to the information they carry. One such example from our dataset is the use of point clouds representing airplanes with varying numbers of engines on the wings. In this section, we analyze the saliency maps generated by our method to understand how different numbers of engines on the wings affect the output class score. These saliency maps are generated using the classification model that was trained on the dataset containing point clouds in the default orientation.

In Figure 10, we have saliency maps for eight input point cloud data instances representing airplanes. The saliency maps are generated using the *absence of feature* mechanism and the segments are generated using the segmentation models (no clustering). We observe that the saliency maps indicate that the wings are the strongest features for examples in the top row (Figure 10a, Figure 10b, Figure 10c, Figure 10d), while the fuselage is the most important feature for the classification model in the bottom row (Figure 10e, Figure 10f, Figure 10g, Figure 10h). A more detailed examination of these samples shows that the major difference between the examples on the top row and bottom row is the number of engines. The examples in the top row have two engines whereas the examples in the bottom row have four engines. The shifting of wings to a selected point in the top four examples leads to the two engines adding minute information to the retained structure as these engines are located very close to the fuselage. However, in the remaining examples, the



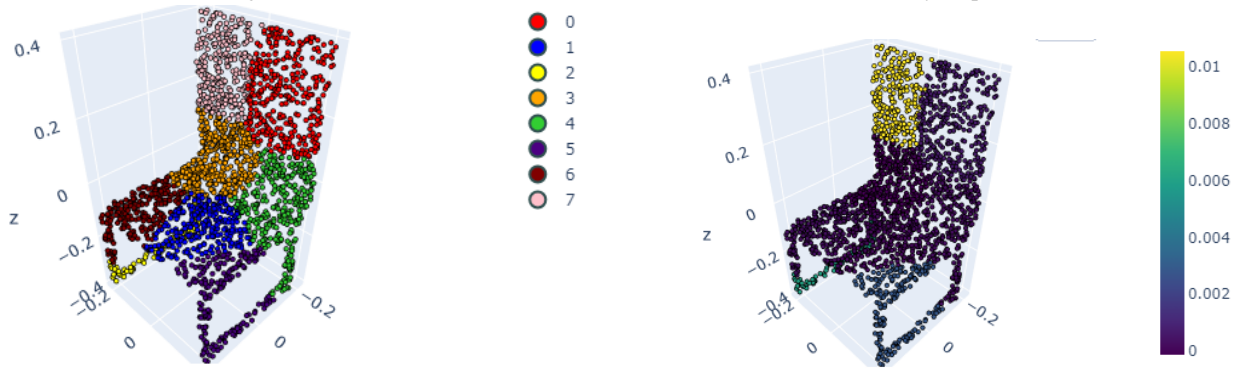
(a) Segments for c=3

(b) Saliency map for c=3



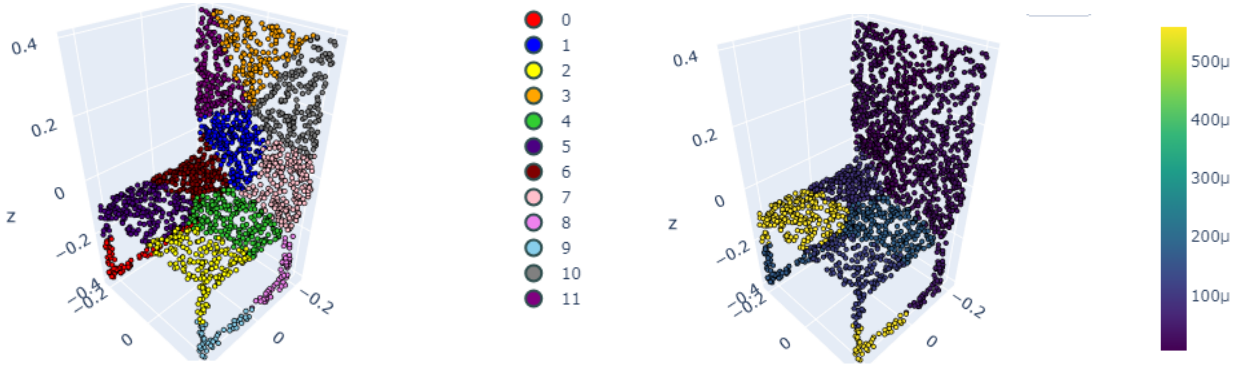
(c) Segments for c=5

(d) Saliency map for c=5



(e) Segments for c=8

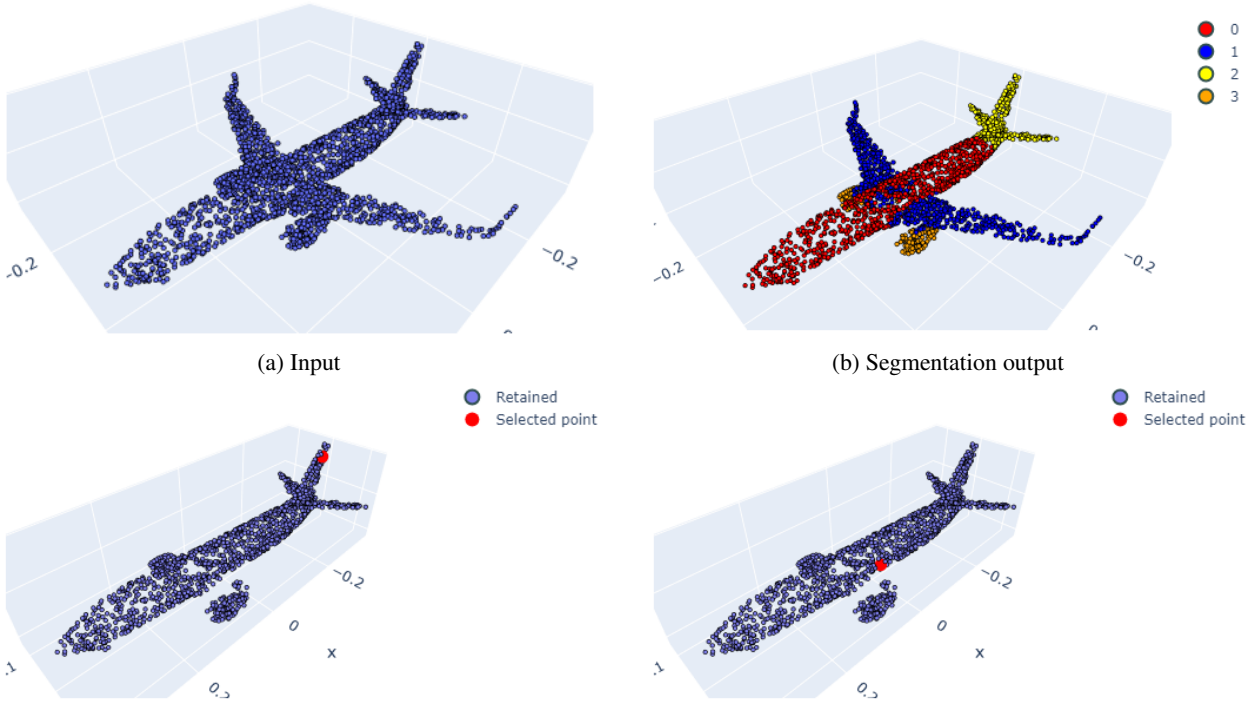
(f) Saliency map for c=8



(g) Segments for c=12

(h) Saliency map for c=12

Figure 8: Saliency maps produced by the *absence of feature* mechanism for clusters produced by KMeans clustering method. *c* represents the number of clusters.



(a) Input

(b) Segmentation output

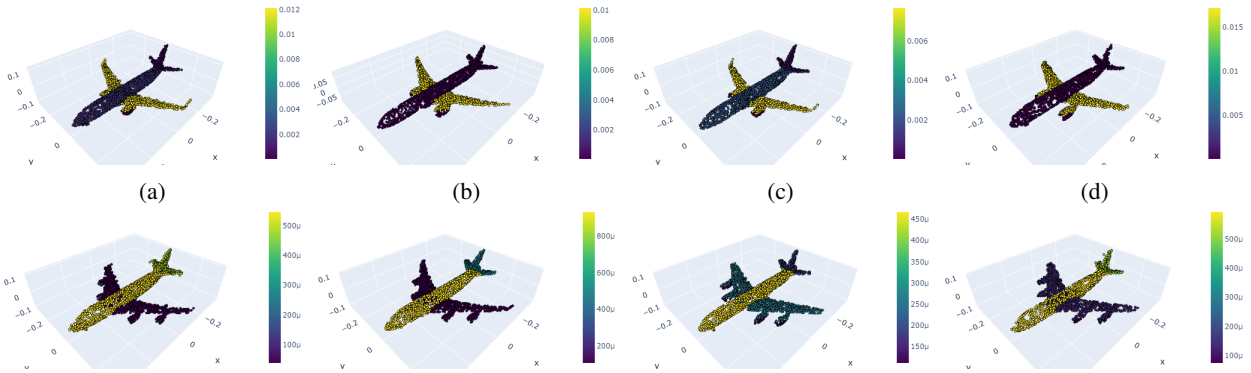
(c) Perturbation using random point 1

(d) Perturbation using random point 2

Figure 9: Perturbation of the input data using random points (marked in (c) and (d)) selected from the retained structure.

shifting of wings to the selected point leads to a structure with two engines on each side of the fuselage. This captures more information with respect to the overall structure of the airplane and thus leads to a lower influence of wings on the output class value. This is also evident in the color scales of the figures that use saliency values to indicate how influential the features of airplanes are. We observe that the magnitude with which wings affect the output class score is significantly higher compared to other features in the top row's examples. However, the presence of two additional engines in the bottom row's examples brings this magnitude down significantly and leads to the fuselage's influence becoming the biggest among all the features.

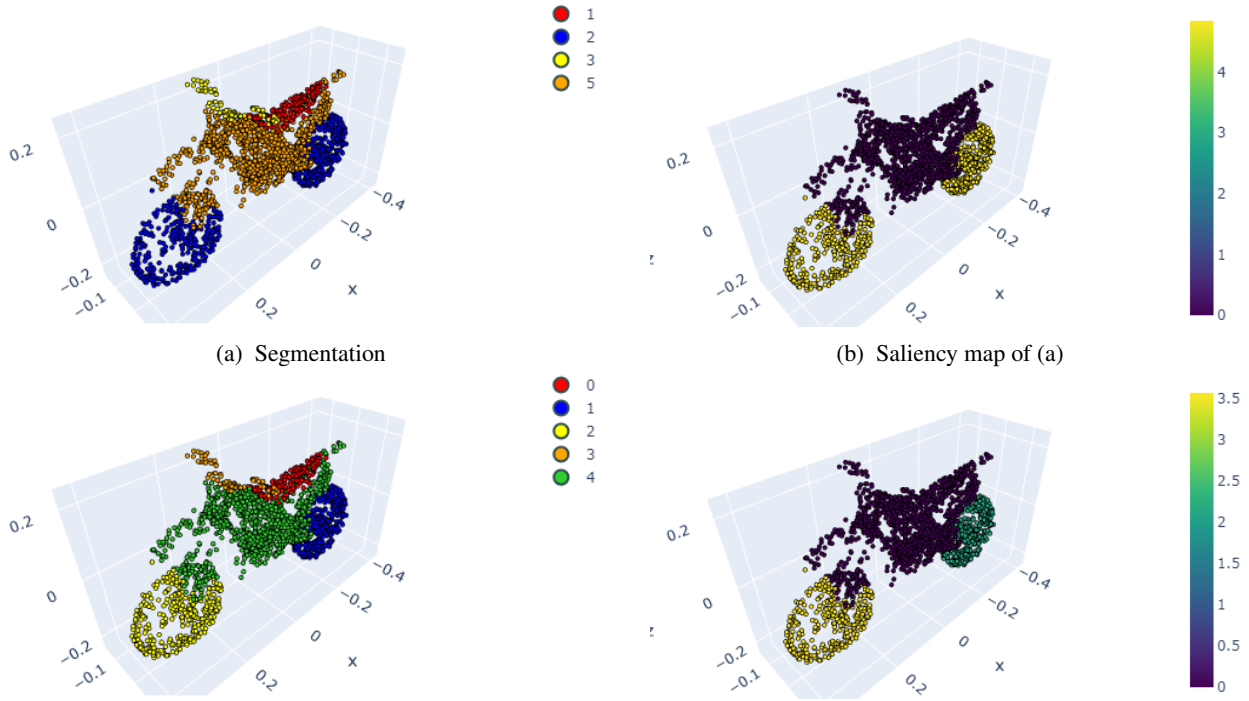
This example gives us an important insight into how the classification model learns to identify and take into consideration different structural information captured by point clouds representing a specific object such as an airplane and make decisions based on this information.



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

(e) (f) (g) (h)

Figure 10: Saliency maps of point clouds representing airplanes using the *absence of feature* mechanism.



(a) Segmentation

(b) Saliency map of (a)

(c) Segmentation+Clustering

(d) Saliency map of (c)

Figure 11: Saliency maps of point clouds representing motorbikes using the *absence of feature* mechanism.

4.4 Segmentation+Clustering: Use case

As described in subsubsection 3.1.2, the Segmentation+Clustering mechanism allows the users to analyze a classification model by taking individual features into account instead of using a set of similar features as one segment. This is useful in understanding the contribution of individual features because a set of features can provide more structural information for the classification model compared to individual features. For example, a set of wheels in a motorbike or car captures more information than a single wheel. We analyze an example to see if this is reflected in the saliency maps produced by our methods.

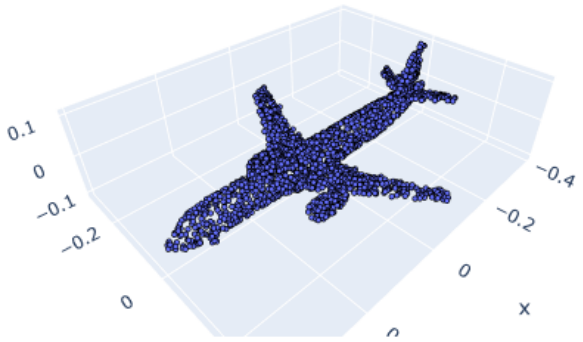
Figure 11 shows an example of saliency maps generated for point cloud data representing a motorbike. The division of input data into multiple clusters is performed using both segmentation and segmentation+clustering mechanisms. We observe that the wheels are clustered as one segment in Figure 11a while the segmentation+clustering mechanism manages to cluster them as separate segments as shown in Figure 11c. We observe that the saliency map generated using the segmentation mechanism indicates high importance for wheels. However, it does not provide any information regarding which wheel is more influential. This is addressed by the segmentation+clustering mechanism which enables us to introduce more specific perturbations into the input data using individual wheels. The saliency map generated using this mechanism is shown in Figure 11d. It shows the front wheel to have more influence on the output class score compared to the rear wheel. This is useful mainly because the wheels of the motorbike are not identical and their locations with respect to the remaining features in the input point clouds are also different. Therefore, these wheels are expected to have different levels of influence on the output class score which is highlighted in Figure 11d.

4.5 Performance analysis

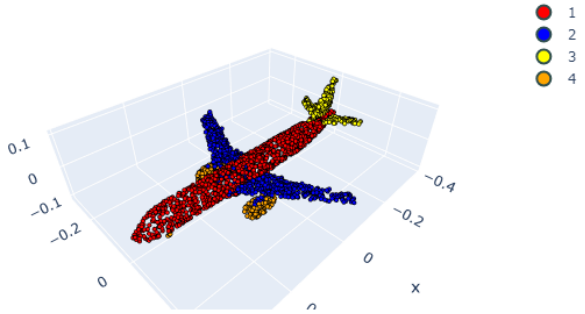
We analyze the performance of our methods using the ground truth of the segmentation task and noisy point cloud data as the input for the classification model in the pipeline.

4.5.1 Ground truth

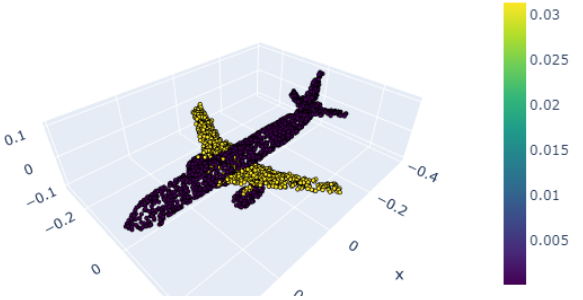
This analysis corresponds to the saliency maps generated by our method based on the segments present in the ground truth (GT) segmentation. The ground truth is used in the third stage of our XAI pipeline which is used to perturb the input point cloud data. Figure 12 shows an example of the saliency map generated for the ground truth of an



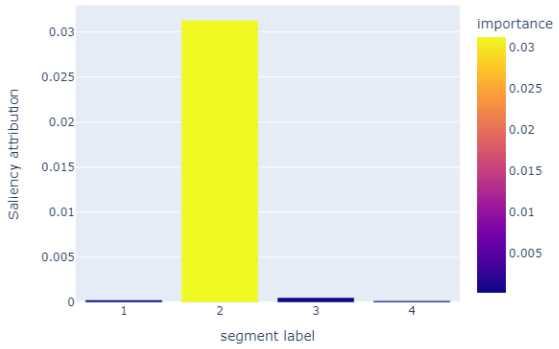
(a) Input



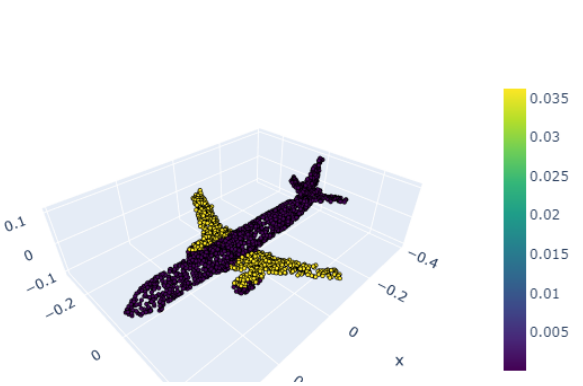
(b) Segmentation GT



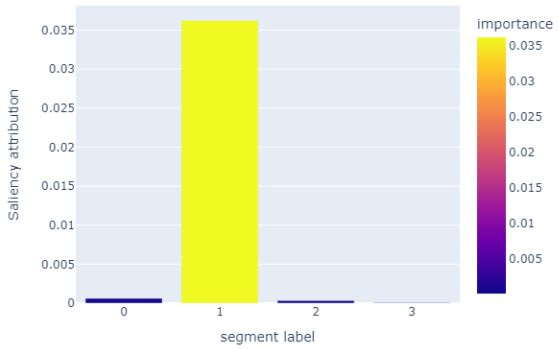
(c) Saliency map with GT



(d) Saliency attributions



(e) Saliency map without GT

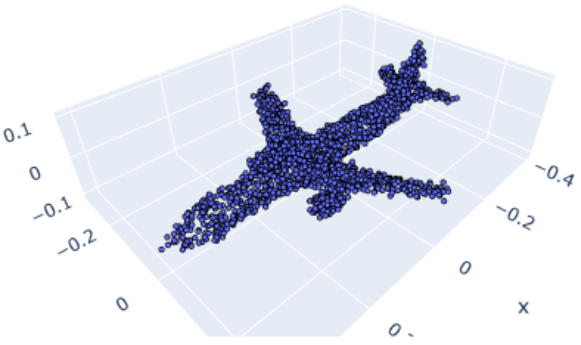


(f) Saliency attributions

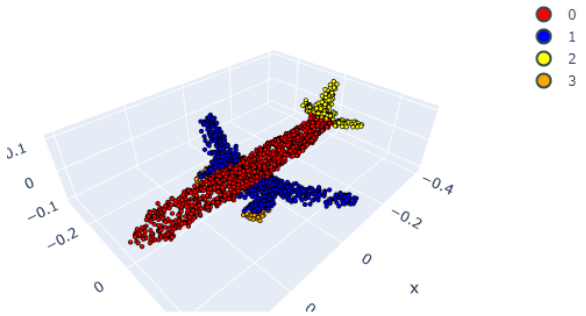
Figure 12: Saliency maps produced with ((c) & (d)) and without ((e) & (f)) using the segmentation ground truth labels using the *absence of feature* mechanism. Note: Refer to the segmentation ground truth figure (b) for corresponding parts represented along the x-axis in the bar plot (d) & (f).

input instance representing an airplane. We use this scenario because the ground truth is the "perfect output" of the segmentation model. In other words, ground truth would be the output of the segmentation model if it had 100% accuracy. Therefore, it is important to analyze the performance of segmentation models in generating the saliency maps as they are a central part of our proposed method.

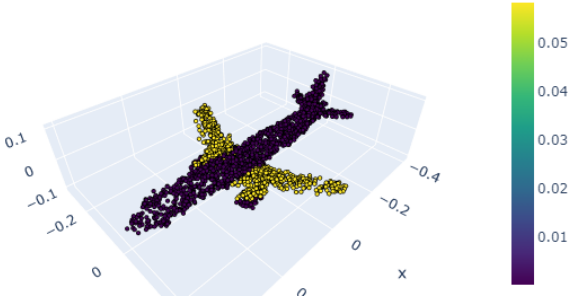
An example of the saliency maps produced using ground truth and the output of the segmentation model is shown in Figure 12. We compare the saliency maps to analyze how the inaccuracy of a segmentation model affects the saliency attributions of the segments in the input data. We observe that the segmentation model manages to produce saliency



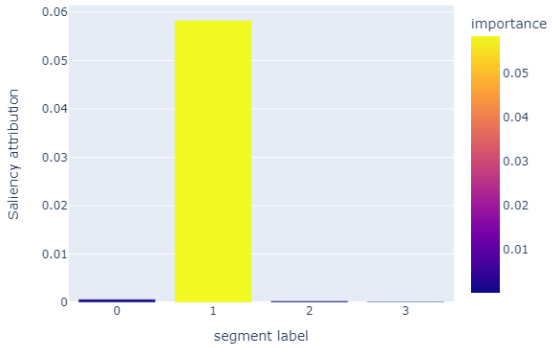
(a) Noisy input



(b) Segmentation output



(c) Saliency map



(d) Saliency attributions of segments.

Figure 13: Noisy input with 5% noise and its corresponding saliency attributions produced by our proposed method.Note: Refer to the 'segmentation output' figure (b) for corresponding parts represented along the x-axis in the bar plot (d).

maps ((e) & (f)) similar to those produced using the segmentation ground truth data ((c) & (d)). This highlights the ability of segmentation models to generate meaningful segments with high accuracy, which leads to the production of meaningful saliency maps.

4.5.2 Noisy input

During the training process, a classification model learns to produce a desired output by tuning its parameters based on the input instance provided and its corresponding ground truth. At the end of the training process, the model parameters are tuned well enough to produce the desired output for a subset of the training dataset (assuming the model does not reach 100% accuracy). However, to analyze the model more effectively, it is important to test its performance on input instances that the model has not seen during its training process.

One of the most common mechanisms of generating new examples for testing AI models is by adding noise to the available data instances. We use this mechanism to test the classification model as well as our XAI mechanism. We introduce noise into the input data instances and analyze the saliency maps generated. An example of this analysis is shown in Figure 13. We observe that our method produces similar saliency attributions for a noisy input data instance that is generated by adding 5% noise to the actual input data (see Figure 12e and Figure 12f). We observed that the method produces saliency maps with minute variations up to 10% noise level. However, higher levels of noise magnitude lead to bigger changes in the input data and, therefore, lead to the generation of incorrect segmentation, leading to incorrect saliency maps. This indicates that the classification model is robust to noise in the input data and also highlights the performance of segmentation models that are a major part of our proposed method.

4.6 Limitations

One of the limitations of our proposed method is associated with the use of a segmentation algorithm. This limitation is the possible inaccuracy of the segmentation model that is trained on the point cloud data. The segmentation models

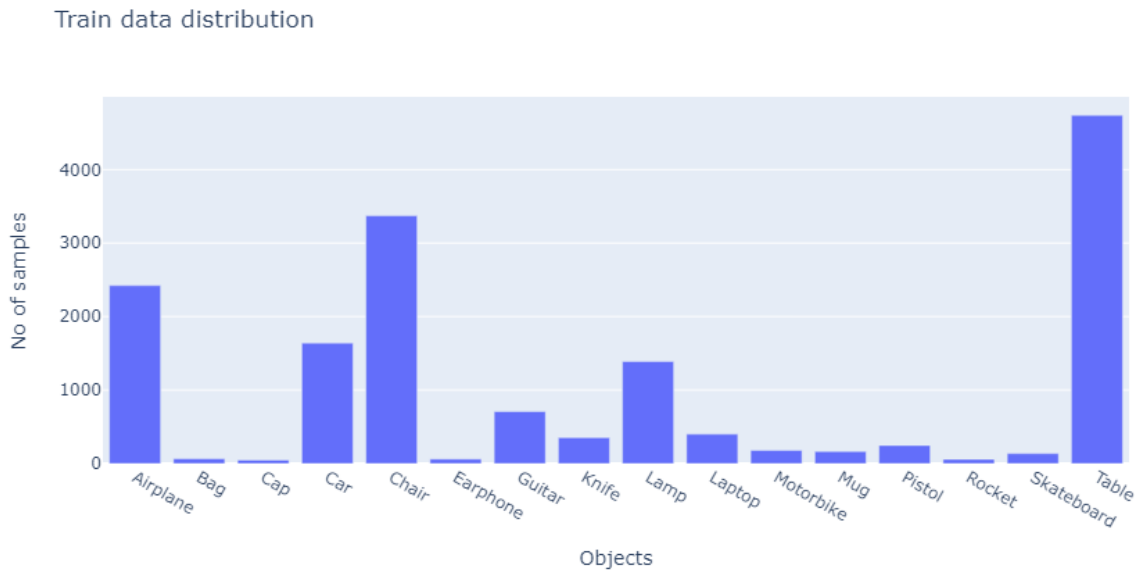


Figure 14: Distribution of the training data.

used in this work have accuracy values in the range of 65%-80%. This is mainly due to two reasons. The first reason corresponds to the imbalance in the dataset. The dataset contains varying numbers of samples representing individual 3D model types with some of them having less than 100 samples as shown in Figure 14. This makes learning difficult for the segmentation models. The second reason corresponds to the structural information associated with these 3D models. Some point clouds represent simple 3D models such as chairs, tables, and laptops which make it easier for the corresponding segmentation models to learn the segmentation task. However, point clouds representing complex 3D models such as motorbikes, cars and airplanes make learning more difficult for the AI models.

The second limitation is also associated with the dataset. This limitation is the requirement of a labeled dataset for segmentation in case we decide to add another 3D object or category to the classification task. This is due to the use of segmentation models that are trained for the segmentation of point clouds of specific categories.

The last limitation corresponds to the dependency of the XAI method on the output label of the classification model when working without human input in the pipeline. Currently, the method uses the classification output to find the corresponding segmentation model. However, when a classification model incorrectly classifies the input data, it will lead to selecting a segmentation model that is inappropriate for the input data. However, human-in-the-loop can easily resolve this problem with the user selecting the segmentation model based on the input point cloud data.

5 Conclusion

In this paper, we proposed a segmentation-based XAI method for understanding the decision-making process of classification models working on point cloud data. The proposed method is based on a perturbation mechanism. It specifically uses meaningful segments to introduce perturbations and thus, produces more meaningful saliency maps. We used two types of perturbation mechanisms to generate explanations with two different perspectives. This allows users to gain better insight into the decision-making process of a classification model and the information carried by each segment in the input data. For the segmentation task, we proposed two mechanisms that leverage segmentation models and clustering algorithms to generate saliency maps. We also proposed a new point-shifting mechanism for the perturbation, to improve explainability. Applying the method to several representative examples, we highlighted the usefulness of our proposed method and analyzed its performance using different input data instances. The proposed method is model-agnostic and therefore can be used to explain any classification model working on point cloud data, irrespective of the model architecture.

Our future work will be to address the limitations mentioned in subsection 4.6.

References

[1] Muhammed Enes Atik, Zaide Duran, and Dursun Zafer Seker. Explainable artificial intelligence for machine learning-based photogrammetric point cloud classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17:5834–5846, 2024. doi:10.1109/JSTARS.2024.3370159.

[2] Elena Camuffo, Daniele Mari, and Simone Milani. Recent advancements in learning algorithms for point clouds: An updated overview. *Sensors*, 22(4), 2022. doi:10.3390/s22041357.

[3] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. URL: http://arxiv.org/abs/1512.03012.

[4] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[5] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3D point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12):4338–4364, 2021. doi:10.1109/TPAMI.2020.3005434.

[6] Eleni Lavasa, Christos Chadoulos, Athanasios Siouras, Ainhoa Etxabarri Llana, Silvia Rodríguez Del Rey, Theodore Dalamagas, and Serafeim Moustakidis. *Toward Explainable Metrology 4.0: Utilizing Explainable AI to Predict the Pointwise Accuracy of Laser Scanning Devices in Industrial Manufacturing*, pages 479–501. Springer Nature Switzerland, Cham, 2024. doi:10.1007/978-3-031-46452-2_27.

[7] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 4765–4774, 2017. URL: https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

[8] James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[9] Francesca Matrone, Marina Paolanti, Andrea Felicetti, Massimo Martini, and Roberto Pierdicca. BubblEX: An explainable deep learning framework for point-cloud classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:6571–6587, 2022. doi:10.1109/JSTARS.2022.3195200.

[10] Siqi Miao, Yunan Luo, Mia Liu, and Pan Li. Interpretable geometric deep learning via learnable randomness injection, 2023. arXiv:2210.16966, doi:10.48550/arXiv.2210.16966.

[11] Raju Ningappa Mulawade, Christoph Garth, and Alexander Wiebel. Explainable artificial intelligence (xai) for methods working on point cloud data: A survey. *IEEE Access*, 12:146830–146851, 2024. doi:10.1109/ACCESS.2024.3472872.

[12] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. doi:10.1109/CVPR.2017.16.

[13] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. doi:10.1145/2939672.2939778.

[14] Anna Saranti, Bastian Pfeifer, Christoph Gollob, Karl Stampfer, and Andreas Holzinger. From 3d point-cloud data to explainable geometric deep learning: State-of-the-art and future challenges. *WIREs Data Mining and Knowledge Discovery*, 14(6):e1554, 2024. doi:10.1002/widm.1554.

[15] L. S. Shapley. A value for n-person games. pages 307–318, 1953. doi:doi:10.1515/9781400881970-018.

[16] Wen Shen, Qihan Ren, Dongrui Liu, and Quanshi Zhang. Interpreting representation quality of DNNs for 3D point cloud processing. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 8857–8870. Curran Associates, Inc., 2021. doi:10.48550/arXiv.2111.03549.

[17] Wen Shen, Zhihua Wei, Qihan Ren, Binbin Zhang, Shikun Huang, Jiaqi Fan, and Quanshi Zhang. Interpretable rotation-equivariant quaternion neural networks for 3D point cloud processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3290–3304, 2024. doi:10.1109/TPAMI.2023.3346383.

[18] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017. doi:10.48550/arXiv.1703.01365.

[19] Saeid Asgari Taghanaki, Kaveh Hassani, Pradeep Kumar Jayaraman, Amir Hosein Khasahmadi, and Tonya Custis. PointMask: Towards interpretable and bias-resilient point cloud processing, 2020. arXiv:2007.04525, doi:10.48550/arXiv.2007.04525.

[20] Hanxiao Tan. Fractual projection forest: Fast and explainable point cloud classifier. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4240–4249, January 2023. doi:10.1109/WACV56688.2023.00422.

[21] Hanxiao Tan. Visualizing global explanations of point cloud DNNs. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4730–4739, 2023. doi:10.1109/WACV56688.2023.00472.

[22] Hanxiao Tan. Flow AM: Generating point cloud global explanations by latent alignment. 2024. arXiv:2404.18760, doi:10.48550/arXiv.2404.18760.

[23] Hanxiao Tan and Helena Kotthaus. Surrogate model-based explainability methods for point cloud NNs. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2927–2936, 2022. doi:10.1109/WACV51458.2022.00298.

[24] Hanxiao Tan and Helena Kotthaus. Explainability-aware one point attack for point cloud neural networks. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4570–4579, 2023. doi:10.1109/WACV56688.2023.00456.

[25] F.M. Verburg. Exploring explainability and robustness of point cloud segmentation deep learning model by visualization, February 2022. URL: http://essay.utwente.nl/89440/.

[26] Li Yi, Vladimir G Kim, Duygu Ceylan, I Shen, Mengyan Yan, Hao Su, ARCewu Lu, Qixing Huang, Alla Sheffer, Leonidas Guibas, et al. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (TOG)*, 35(6):210, 2016.

[27] Min Zhang, Haoxuan You, Pranav Kadam, Shan Liu, and C.-C. Jay Kuo. PointHop: An explainable machine learning method for point cloud classification. *IEEE Transactions on Multimedia*, 22(7):1744–1755, 2020. doi:10.1109/TMM.2019.2963592.

[28] Tianhang Zheng, Changyou Chen, Junsong Yuan, Bo Li, and Kui Ren. Pointcloud saliency maps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. doi:10.48550/arXiv.1812.01687.