2D Manifold Topology Representation Learning

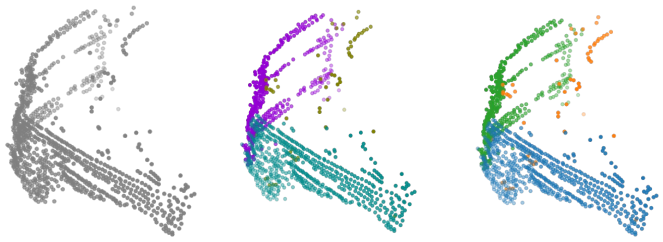
Magdy Mahmoud
Chair of Robotics, Artificial Intelligence and Real-time Systems
TUM School of Computation, Information, and Technology
Munich, Germany
magdy.mahmoud@tum.de

***Abstract*—Autonomous driving systems rely on precise perception of the environment, particularly in detecting and understanding 3D objects for safe navigation. Despite advancements, processing 3D data from LiDAR sensors remains challenging. Recent research has shown promise in leveraging point clouds for various tasks, yet surface features are often overlooked. Our research presents a novel semi-supervised vehicle part segmentation method that exploits surface features and unsupervised clustering. We propose a two-stage approach: learning surface features via classification and unsupervised clustering based on combined features and estimated surface normals. Evaluation on a manually labeled subset of the KITTI dataset demonstrates the effectiveness of our approach, representing a significant advancement in 3D perception for autonomous driving. By enhancing the richness of available data through surface feature extraction, our method yields several advantages for various 3D tasks, such as enhancing 3D detection, segmentation, classification, and pose estimation. By providing richer and more accurate environmental information, our approach contributes to the advancement of these crucial aspects in autonomous driving systems.**

I. Introduction

Autonomous driving systems rely heavily on accurate perception of the surrounding environment to navigate safely and efficiently. Central to this perception is the ability to detect and understand 3D objects within the scene. From identifying pedestrians and other vehicles to recognizing road signs and obstacles, robust 3D perception methods is paramount for the success of autonomous vehicles. Despite advancements in sensor technology, processing the wealth of 3D data captured by LiDAR sensors remains a formidable challenge. Traditional methods often struggle to extract meaningful information from the complex and high-dimensional point clouds generated by these sensors. Recent research has shown promising results in leveraging point clouds for various tasks, with approaches such as PointNet [1], PointNet++ [2], DGCNN [3], PointNext [4] and RepSurf [5] demonstrating the potential of point cloud processing in object detection, segmentation, and classification.

Moreover, surface features and reconstruction play a pivotal role in scene understanding and various 3D tasks. Despite this, existing methodologies often overlook surface information, relying instead on geometric and volumetric representations of objects, which provide limited information about finer details within objects. However, for critical situations like collision avoidance, understanding vehicle parts (e.g., hood vs. door) is essential for informed decision-making. For example, Autolabeling3D [6] leverages differentiable rendering of Signed Distance Function (SDF) shape priors to reconstruct object surfaces accurately. By incorporating SDF shape priors, the method achieves high-fidelity surface reconstruction, allowing for precise delineation of object boundaries in 3D space. This reconstructed surface information is then utilized to generate annotations for 3D bounding boxes, which demonstrate comparable quality to the annotations provided in the original KITTI [7] dataset.



(a) Input point cloud (b) Clustering (c) Matching

Fig. 1: Visualization of our vehicle part segmentation pipeline: (a) Input point cloud, (b) Clustering of surface features, and (c) Matching of clustered parts.

Our research takes inspiration from these seminal works and aims to build upon their advancements. Recognizing the limitations in capturing local geometric structures, we introduce a novel method that harnesses surface features within 3D point clouds, thereby enhancing the richness of available data for scene analysis. Central to our methodology is the exploration of the synergies between 2D manifold topology and 3D point clouds. By modeling surfaces as 2D manifolds, we can exploit various mathematical tools and techniques to extract valuable information from the data. This approach not only promises to advance the state-of-the-art in 3D object detection but also holds potential for improving other 3D tasks, such as object segmentation, classification, and pose estimation.

Our approach introduces a self-supervised learning paradigm, wherein the model is trained using points within ground truth bounding boxes. This strategy allows the model to learn discriminative features directly from the data without relying on manually labeled annotations for every data point. To assess the performance of our method, we manually annotate a subset of the KITTI dataset, providing labeled data for evaluation. By combining surface feature

extraction, surface normal estimation, feature clustering, and cluster matching, we demonstrate the effectiveness of our semi-supervised part-segmentation approach through extensive experiments on the manually annotated dataset. We compare multiple combinations of surface-feature extractors and unsupervised clustering algorithms to identify the optimal algorithmic stack for achieving accurate and robust part segmentation. Overall, our method represents a significant step forward in leveraging surface features for enhanced 3D perception in autonomous driving and related domains. An example of our vehicle part segmentation pipeline is shown in Fig. 1.

II. Related Works

A. Point Cloud Segmentation

Point cloud segmentation enhances understanding of 3D scenes and represents a significant research direction in autonomous driving. PointNet [1] and PointNet++ [2] pioneer multi-layer perceptron networks for processing unstructured point clouds. Subsequently, more methods have been proposed, such as [8]–[10], exploring using convolution neural networks to learn point features. DGCNN [3] proposes using the EdgeConv module to obtain edge features from the KNN-based local graphs. PointNext [4] based on [2] enlarges the receptive fields while keeping rich semantic features. One of the most recent works, RepSurf [5], exploits triangle-based and multi-surface representation for segmentation. These methods achieve promising performance on part segmentation of indoor objects or semantic segmentation in outdoor scenarios. However, the application of part segmentation on objects in the autonomous driving domain has been under-explored, while it is vital for improving the performance of perception systems.

B. Semi-Supervised Learning

Semi-supervised learning allows combining labeled and unlabeled data to perform certain tasks [11], which has significant advantages in eliminating the data-hungry problem and keeping task performance. [12] proposed a semi-supervised learning approach utilizing GANs, which leads to the generation of more effective classifiers and higher-quality samples. FixMatch [13] improved the performance by combining consistency regularization and pseudo-labeling. [14] explores compacting latent space clustering for semi-supervised learning. Considering the benefits of semi-supervised learning, we exploit using it to facilitate part segmentation tasks in autonomous driving.

C. Unsupervised Clustering

Clustering aims to partition similar data into the same group and push dissimilar data far from each other [15], which is widely used in various computer vision tasks, such as anomaly detection, segmentation, and many more. Usually, as no predefined categories provide prior knowledge, it is also called unsupervised clustering. K-Means [16], as one of the most popular algorithms, partitions the input data into K distinct, non-overlapping clusters. K is the predefined number of clusters. Compared to K-Means, the Gaussian Mixture Model [17] follows a probabilistic approach, assuming the data is generated from several Gaussian distributions. In our study, we verify ten types of clustering algorithms to find the most optimal combination for part segmentation and learning surface features.

D. Surface Reconstruction

Surface reconstruction is pivotal for understanding 3D scenes and objects in the domain of autonomous driving. Foundational works like Localization and Mapping using Instance-specific Mesh Models [18] proposes a method for building semantic maps, containing object poses and shapes, using a monocular camera. Similarly, On the Road to Large-Scale 3D Monocular Scene Reconstruction using Deep Implicit Functions [19] introduces a framework for large-scale 3D monocular scene reconstruction using deep implicit functions, enabling detailed scene reconstruction from single-camera setups. Recent advancements, like Mending Neural Implicit Modeling for 3D Vehicle Reconstruction in the Wild [20] focuses on improving neural implicit modeling for 3D vehicle reconstruction in challenging real-world conditions, enhancing the robustness and accuracy of 3D vehicle reconstruction through novel training strategies and architecture modifications.

Other works explores combining surface reconstruction with other 3D tasks. For example, DOPS: Learning to Detect 3D Objects and Predict their 3D Shapes [21] presents a framework for simultaneously detecting 3D objects and predicting their shapes from RGB images, leveraging deep learning and geometric reasoning for accurate object detection and shape prediction. On the other hand, Autolabeling3D Objects with Differentiable Rendering of SDF Shape Priors [6] introduces a technique for autolabeling 3D objects using differentiable rendering of signed distance function (SDF) shape priors. By leveraging shape priors encoded in SDF representations, this method enables automatic annotation of 3D objects, contributing to more efficient surface reconstruction pipelines. Subsequently, Online Adaptation for Implicit Object Tracking and Shape Reconstruction in the Wild [22] addresses the challenge of online adaptation for implicit object tracking and shape reconstruction in real-world environments, enhancing the robustness of object tracking and shape reconstruction in dynamic scenes.

More recent works such as MV-DeepSDF [23], proposes a framework for implicit modeling using multi-sweep point clouds. By leveraging multiple sweeps, MV-DeepSDF achieves accurate reconstruction of 3D vehicles in autonomous driving scenarios, contributing to improved perception systems. These works collectively advance the state-of-the-art in surface reconstruction for autonomous driving applications, addressing challenges such as robustness, accuracy, and efficiency. Our proposed approach builds upon these contributions, leveraging insights from surface reconstruction literature to enhance unsupervised part segmentation in point cloud data, with a focus on vehicle surfaces.

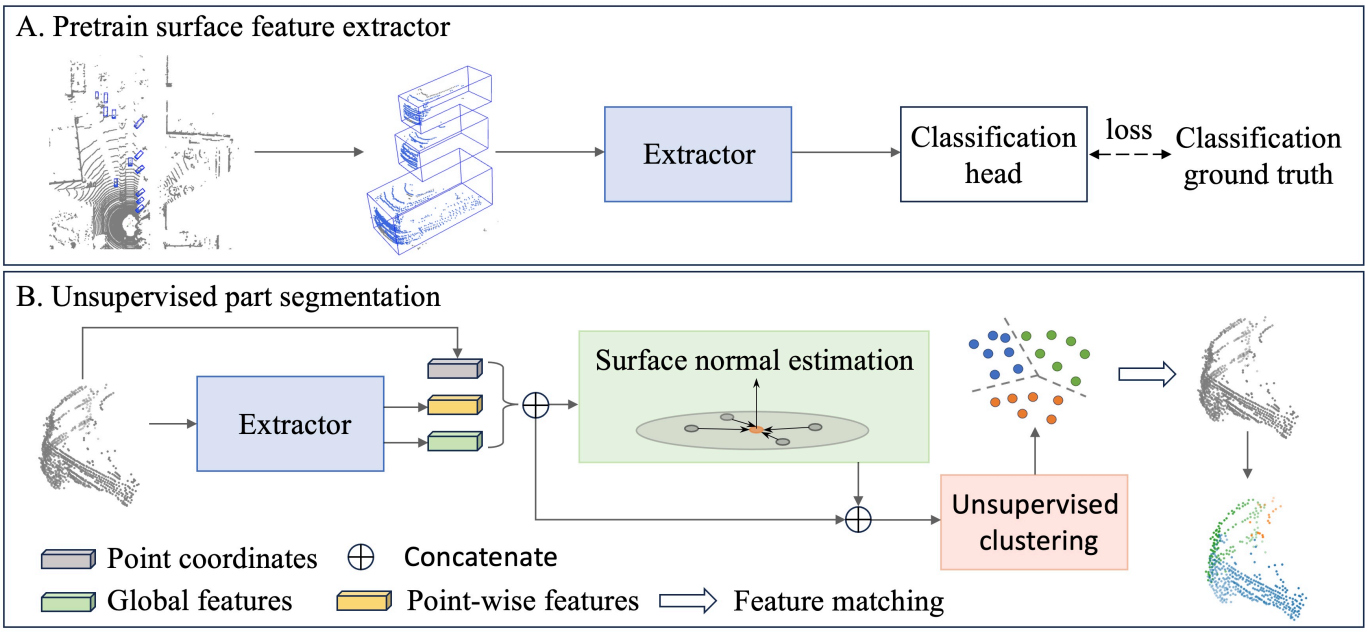


Fig. 2: Overview of the proposed semi-supervised part segmentation pipeline, showcasing the two-stage process: (A) initial surface feature learning via classification, (B) followed by surface normal estimation and unsupervised clustering.

III. Methodology

A. Overview

Our proposed semi-supervised part segmentation pipeline, illustrated in Fig. 2, follows a two-stage approach. First, a model is trained to learn the surface features of the point clouds using the classification label as supervision. These features are then extracted from the trained model and used with the original point cloud to estimate the surface normals, as explained in more detail in Section III-B. Second, these features, along with the surface normals, are then combined for unsupervised clustering, which are then matched with ground truth clusters for evaluation, as discussed in more detail in Section III-C.

B. Surface Features Learning

Our method aims to extract surface features from 3D point clouds without the need for per-point surface annotation. To achieve this, we leverage the readily available classification labels associated with each object in the KITTI dataset. Unlike per-point annotation, classification labels are easily obtainable and serve as effective proxies for learning discriminative features. As depicted in Fig. 2 part A, our approach begins by extracting the bounding boxes of objects in the input point cloud, along with their respective classification labels. While this extraction process can be facilitated using 3D object detection techniques like PointPillars [24], we opted to use the ground truth annotations provided by the KITTI dataset solely for prototyping and testing the effectiveness of our approach.

We utilize a learning-based network to extract discriminative features from each object point cloud. This network is trained using the classification labels associated with each object, providing valuable supervision for feature learning. Through this training process, the network learns to capture surface features at both the individual point level and the broader global scale. For this task, we use PointNext [4] and Repsurf [5], which follow the standard encoder-decoder architecture [25]. Since we train this network for classification only, the decoder part is replaced by a multilayer perceptron (MLP). During training, we employ a smooth cross-entropy loss function, defined as:

$$l(y,p) = -(y\log(p) + (1-y)\log(1-p)) \tag{1}$$

where p is the predicted possibility and y is the indicator.

By leveraging classification labels and innovative network architectures, our method effectively learns to capture surface features from 3D point clouds. We then extract the learned features from the first layer of the network with the corresponding point coordinate. The combination can be described as $\mathbf{f}_i^{'} = (\mathbf{f}_i, x_i, y_i, z_i)$, where $\mathbf{f}_i \in \mathbb{R}^{1\times d}$ is the point-wise feature of point $i \in \{1,\ldots,n\}$, and (x_i, y_i, z_i) are its coordinate. Additionally, we combine the global features $\mathbf{g} \in \mathbb{R}^{1\times d^{'}}$ from the last layer of the extractor with $\mathbf{f}^{'}$. This fusion of features ensures a comprehensive representation of the point cloud, capturing both local and global information.

Before proceeding with the clustering process, we further enhance our features by incorporating additional geometric cues to enrich the representation of the object's geometry. To achieve this, we estimate the surface normals using the concatenated features from the first step. Surface normal estimation involves computing the covariance matrix of the k nearest neighbors for each concatenated feature using the k-nearest neighbor (KNN) approach. Singular Value Decom-

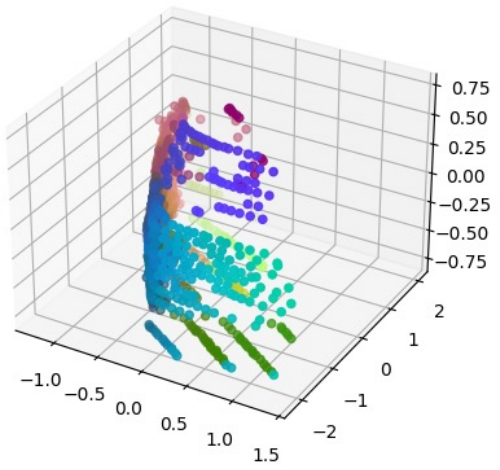


Fig. 3: We illustrate the combined features of a car via UMAP [26]. Points with the same color indicate similar features, such as the front in blue and the back in brown.

position (SVD) is then applied to extract the normals from the direction of minor variance. Finally, the normals are normalized and oriented outward from the surface before being combined with the features obtained in the previous step. To gain insight into the distribution of the combined features and observe the emergence of distinct clusters, we employ dimension reduction techniques such as Uniform Manifold Approximation and Projection (UMAP) [26] as illustrated in Fig. 3. This visualization approach offers a clear indication of the features beginning to form different clusters.

C. Features clustering

After obtaining the combined features and estimated surface normals from the previous step, our next objective is to cluster these features to effectively separate different surface characteristics. The objective here is to group similar features together to identify coherent surface characteristics. We leverage various unsupervised clustering algorithms for this task, including but not limited to: K-Means [16], Gaussian Mixture Model [17] and Spectral Clustering [27]. These algorithms partition the feature space into clusters based on their inherent structure, allowing us to identify and delineate different parts of the object's surface.

Following the clustering of features, our next step is to align the clusters generated by our algorithm with the ground truth clusters in the labeled dataset. This alignment process is crucial as it allows us to assess the effectiveness of our method. First, as outlined in Alg. 1, we construct a cost matrix by computing the Intersection over Union (IoU) for each element of the predicted clusters with each class of the ground truth data to quantify the similarity between the clusters. Once the cost matrix is obtained, we apply the Hungarian algorithm to find the optimal assignments. The Hungarian algorithm efficiently assigns predicted clusters to ground truth clusters based on minimizing the total cost. This meticulous process ensures that each predicted cluster is matched with the most suitable ground truth cluster, facilitating a comprehensive evaluation of our method.

Algorithm 1 Calculate Cost Matrix

Require: Unsupervised labels U, Ground truth labels G, Number of unsupervised clusters n, Number of ground truth labels m
Ensure: Cost matrix C

function IoU(A, B)
  return $\frac{|A \cap B|}{|A \cup B|}$
end function
$C \leftarrow$ matrix of zeros of size $n \times m$
for $i = 1$ to n **do**
  for $j = 1$ to m **do**
    $U_i \leftarrow$ set of points in U with label i
    $G_j \leftarrow$ set of points in G with label j
    if G_j is not empty **then**
      $iou \leftarrow \text{IoU}(U_i, G_j)$
      $C_{i,j} \leftarrow 1 - iou$
    end if
  end for
end for
return C

IV. Experiments and Results

In this section, we introduce the datasets and models used in our experiments. Then, we explain the evaluation metrics and experimental setup and analyze both quantitative and qualitative results in depth.

A. Datasets

KITTI. As one of the pioneering datasets in the domain of autonomous driving and renowned for its diverse urban scenarios and rich annotations [28], KITTI 3D object detection dataset [7] primarily focuses on classes such as cars, pedestrians, and cyclists. The KITTI dataset comprises 7,481 annotated images and associated LiDAR point cloud scans. It provides a comprehensive view with approximately 200,000 labeled object instances in total. In our study, we utilize the classification labels and points within the ground truth bounding boxes for the training of feature extractors.

Manually Annotated Data. We manually labeled 64 cars from the KITTI dataset to evaluate the efficiency of each semi-supervised network on learning surface features and performing part segmentation. We only annotate car points within a bounding box, and noise and ground points are not marked during the experiments. Every point is assigned to a specific class, representing different parts of the vehicle: front, rear, left, right, or top. To provide a consistent number of points across the labels, we extract cars, including more than 1024 points, which are then down-sampled with farthest point sampling to obtain a uniform sample.

Evaluation Metrics In our experiments, we leverage several widely used metrics [29] to evaluate the part segmentation

performance based on the combination of various feature extractors and clusters. We first utilize mean Intersection over Union (IoU) (see Eq. 2) to measure the overlap between the predicted segmentation and the ground truth.

$$mIoU = \frac{1}{C+1}\sum_{m=0}^{C}\frac{p_{mm}}{\sum_{n=0}^{C}p_{mn}+\sum_{n=0}^{C}p_{nm}-p_{mm}} \quad (2)$$

where C is the number of classes, and p_{mm}, p_{mn}, and p_{nm} represent true positives, false positives, and false negatives, respectively.

Additionally, we also take into account using mean Pixel Accuracy as another evaluation metric. In our case, we calculate the correctly classified points over the total number of points:

$$mPA = \frac{1}{C+1}\sum_{m=0}^{C}\frac{p_{mm}}{\sum_{n=0}^{C}p_{mn}} \quad (3)$$

We also apply F1-Score to do the evaluation, which can be demonstrated as:

$$F1 = \frac{2\sum_{n=0}^{C}p_{mm}}{\sum_{m=0}^{C}(2p_{mm}+p_{nm}+p_{mn})} \quad (4)$$

B. Experimental Setups.

We conduct all experiments using PyTorch 2.0 on one NVIDIA GeForce 4090 (24Gb). All surface-feature extractors are trained with 30 epochs with batch size 2. We use the AdamW optimizer with an initial learning rate of 2e-3 for the training.

Surface Feature Extractors We test two types of feature extractors, PointNext [4] and Repsurf [5], to obtain the surface features from the points within bounding boxes. Based on PointNet++ [2], PointNext follows the hierarchical feature-learning scheme by partitioning point clouds into overlapping local regions at multiple scales and varying resolutions. On the other hand, RepSurf focuses on explicitly depicting the local shape of point clouds, emphasizing surface representation. Both of them follow the encoder-decoder architecture. Because we train the extractor on classification, we replace the decoder with a Multilayer Perceptron (MLP).

Unsupervised Feature Clustering In our experiments, we evaluate ten unsupervised clustering algorithms, which can be categorized into four classes: 1) partitioning methods (K-Means [16] and Gaussian Mixture Model [17]), 2) hierarchical methods (Agglomerative Clustering and BIRCH), 3) density-based method (DBSCAN, OPTICS, and HDBSCAN), 4) graph-based methods (Spectral Clustering [27] and Affinity Propagation), and 5) model-seeking method (Mean Shift). In the main experiments, we set the cluster size n to 3 for those algorithms requiring a fixed number of clusters since three sides of the car are present on average. Furthermore, for a fair comparison, we set top-3 filtering in our evaluation pipeline for the methods with automatically adaptive cluster sizes.

C. Experiment results

In this section, we show and discuss the experiment results of different combinations of surface-feature extractors and unsupervised algorithms on our manually annotated data. The quantitative results of surface-feature extractors PointNext and Repsurf are shown in Tab. I and Tab. II, respectively.

First, we evaluate PointNext on the labeled data regarding the mIoU, mPA, and F1-Score. Overall, almost all approaches with fixed cluster sizes perform around 43% under mIoU and over 60% for mPA, which outperforms the approaches with adaptive cluster sizes in terms of all metrics. For example, K-Means achieves 42.1% on mIoU, which is much higher than Mean Shift (29.3%). Notably, Affinity Propagation, another adaptive clustering method, records the poorest performance across all metrics compared to its counterparts. This discrepancy arises due to the absence of a predetermined target cluster number in adaptive clustering, leading to instances of under-segmentation, as observed in OPTICS, DBSCAN, and HDBSCAN (as shown in Fig. 4).

On the contrary, our proposed PointNext, when combined with the Gaussian Mixture Model, emerges as a standout performer among other algorithms, exhibiting substantial enhancements across all three metrics: mIoU 62.3%, accuracy 80.8%, and 43.6% on F1-Score. As illustrated in the qualitative results within the dashed box in Fig. 4, our approach accurately segments the car into the three targeted parts.

Extractor	Method	mIoU	mPA	F1-Score
PointNext	K-MEANS	42.1	61.0	33.0
	Agglonerative Clustering	41.4	61.6	32.5
	Spectral Clustering	43.7	66.7	32.9
	Affinity Propagation	8.8	30.4	12.6
	Mean Shift	29.3	62.2	22.4
	BIRCH	44.4	65.6	34.2
	OPTICS	24.7	60.5	18.8
	DBSCAN	19.3	58.0	14.6
	HDBSCAN	25.6	58.1	20.2
Ours (PointNext + GMM)		**62.3**	**80.8**	**43.6**

TABLE I: We show the experiment results of PointNext [4] combining with ten unsupervised clustering algorithms in terms of three evaluation metrics, mIoU (%) ↑, mPA(%) ↑, and F1-Score(%) ↑. GMM is the Gaussian Mixture Model.

Furthermore, we investigate the utilization of RepSurf as the surface-feature extractor (Tab. II). Each unsupervised clustering method demonstrates comparable performance to the combination with PointNext when RepSurf serves as the feature extractor. Generally, RepSurf slightly enhances the performance of fixed-size approaches. However, despite being outperformed by PointNext as the extractor, the Gaussian Mixture Model remains the most effective clustering method.

In summary, leveraging algorithms with fixed cluster numbers can yield superior results compared to adaptive clustering methods for effective surface feature learning. Additionally, employing a feature extractor that focuses on detailed geometric information can enhance the capture of valuable surface features.

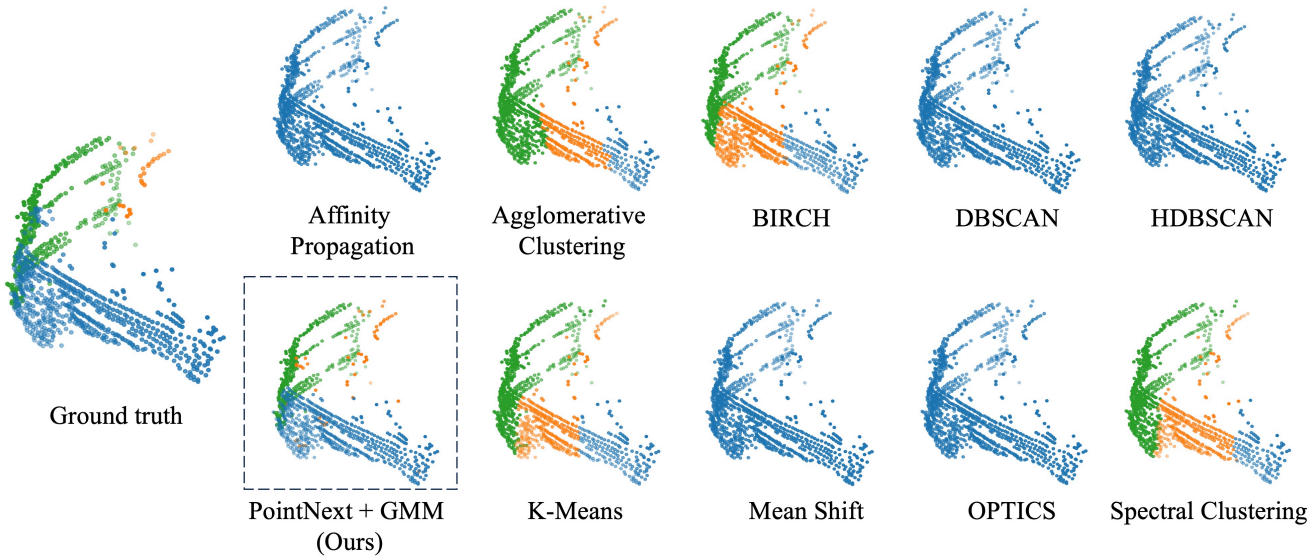


Fig. 4: Qualitative results of part segmentation between different unsupervised clustering algorithms with PointNext as the surface feature extractor. Blue, green, and orange represent the car's right, rear, and top sides for the ground truth points, respectively. The dotted box shows the result of our proposed PointNext combined with the Gaussian Mixture Model (GMM) stack.

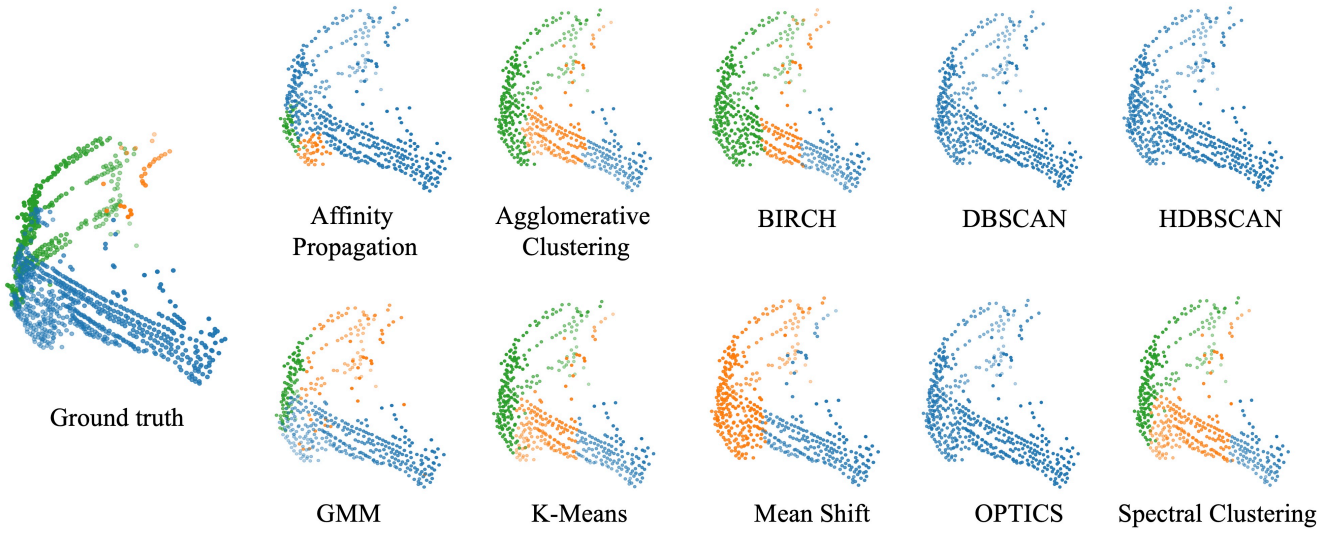


Fig. 5: Qualitative results of part segmentation between different unsupervised clustering algorithms with Repsurf as the surface feature extractor. Blue, green, and orange represent the car's right, rear, and top sides for the ground truth points, respectively.

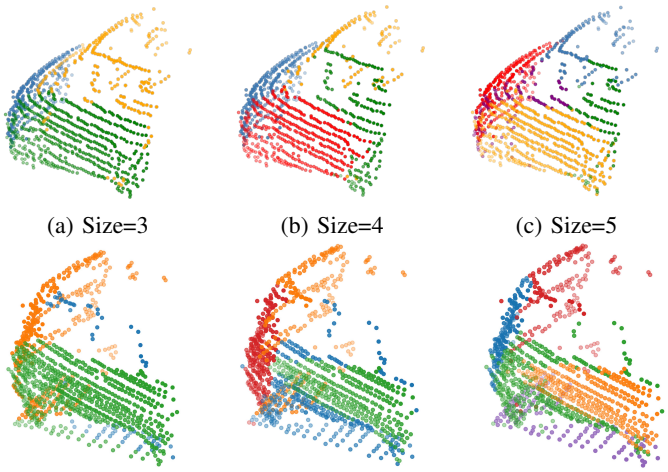
D. Ablation Study

In this section, we study the influence of the cluster size of our proposed approach (PointNext with Gaussian Mixture Model (GMM)) on the segmentation performance. We set 3 different cluster sizes (3, 4, and 5) to qualitatively identify the optimal size for the surface segmentation task. A larger size means the cluster focuses more on details and prefers to segment an object into more parts. We evaluate segmentation performance under two scenarios: without ground truth and including ground truth. The experiment results are depicted in Fig. 6. When we set the cluster to 3, the algorithm effectively segments the car without ground into three parts: right, rear, and top. Conversely, with cluster sizes of 4 or 5, the segmentation results show subdivision of the rear and right sides of the car into multiple sub-parts.

Furthermore, if there are ground points or noise, as shown in the second row, the segmentation performance of the car body looks quite good, even though we struggles to differentiate between object points and ground points. However, when employing larger cluster sizes, the segmentation outcomes degrade even more in comparison to clean input data. Thus,

Extractor	Method	mIoU	mPA	F1-Score
RepSurf	K-MEANS	42.7	60.6	33.6
	Agglonerative Clustering	43.2	62.4	34.1
	Spectral Clustering	44.1	64.4	33.4
	Affinity Propagation	10.9	28.5	14.3
	Mean Shift	33.4	64.0	25.5
	BIRCH	48.1	67.5	36.9
	OPTICS	21.0	57.0	16.2
	DBSCAN	18.6	55.8	14.2
	HDBSCAN	21.9	60.0	17.3
	Gaussian Mixture Model	**58.1**	**76.3**	**41.5**

TABLE II: We show the experiment results of RepSurf [5] combining with ten unsupervised clustering algorithms in terms of three evaluation metrics, mIoU (%) ↑, mPA(%) ↑, and F1-Score(%) ↑.



(a) Size=3 (b) Size=4 (c) Size=5

(d) Size=3 (e) Size=4 (f) Size=5

Fig. 6: Comparison between different cluster sizes of our introduced PointNext [4] combining with GMM approach. The first row ((a), (b), and (c)) shows the segmentation results based on clean input point clouds. The second row exhibits the results considering ground points.

opting for a cluster size of 3 aligns with the goals of our proposed semi-supervised part segmentation pipeline.

V. Conclusion

In conclusion, our research presents a novel methodology for semi-supervised vehicle part segmentation, offering a solution to the challenges posed by conventional methods in capturing intricate local geometric structures. By harnessing surface features alongside unsupervised clustering, we significantly enhance 3D perception, a fundamental requirement for autonomous driving systems. We demonstrate the effectiveness of our approach through extensive experiments on a subset of the KITTI dataset, manually labeled specifically for evaluation purposes. Future endeavors will focus on refining the clustering algorithms further and expanding the application of our methodology to address other critical 3D perception tasks. This advancement holds promise for the development of safer and more efficient autonomous driving systems.

References

[1] C. R. Qi, H. Su, K. Mo, and L. 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*, pp. 652–660, 2017.

[2] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[3] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[4] G. Qian, Y. Li, H. Peng, J. Mai, H. Hammoud, M. Elhoseiny, and B. Ghanem, "Pointnext: Revisiting pointnet++ with improved training and scaling strategies," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23192–23204, 2022.

[5] H. Ran, J. Liu, and C. Wang, "Surface representation for point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18942–18952, 2022.

[6] S. Zakharov, W. Kehl, A. Bhargava, and A. Gaidon, "Autolabeling 3d objects with differentiable rendering of sdf shape priors," 2020.

[7] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*, pp. 3354–3361, IEEE, 2012.

[8] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "Pointcnn: Convolution on x-transformed points," *Advances in neural information processing systems*, vol. 31, 2018.

[9] Y. Xu, T. Fan, M. Xu, L. Zeng, and Y. Qiao, "Spidercnn: Deep learning on point sets with parameterized convolutional filters," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 87–102, 2018.

[10] M. Xu, R. Ding, H. Zhao, and X. Qi, "Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3173–3182, 2021.

[11] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine learning*, vol. 109, no. 2, pp. 373–440, 2020.

[12] J. T. Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," *arXiv preprint arXiv:1511.06390*, 2015.

[13] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *Advances in neural information processing systems*, vol. 33, pp. 596–608, 2020.

[14] K. Kamnitsas, D. Castro, L. Le Folgoc, I. Walker, R. Tanno, D. Rueckert, B. Glocker, A. Criminisi, and A. Nori, "Semi-supervised learning via compact latent space clustering," in *International conference on machine learning*, pp. 2459–2468, PMLR, 2018.

[15] N. Grira, M. Crucianu, and N. Boujemaa, "Unsupervised and semi-supervised clustering: a brief survey," *A review of machine learning techniques for processing multimedia content*, vol. 1, no. 2004, pp. 9–16, 2004.

[16] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, pp. 281–297, Oakland, CA, USA, 1967.

[17] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society: series B (methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[18] Q. Feng, Y. Meng, M. Shan, and N. Atanasov, "Localization and mapping using instance-specific mesh models," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Nov. 2019.

[19] T. Roddick, B. Biggs, D. O. Reino, and R. Cipolla, "On the road to large-scale 3d monocular scene reconstruction using deep implicit functions," in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 2875–2884, 2021.

[20] S. Duggal, Z. Wang, W.-C. Ma, S. Manivasagam, J. Liang, S. Wang, and R. Urtasun, "Mending neural implicit modeling for 3d vehicle reconstruction in the wild," 2021.

[21] M. Najibi, G. Lai, A. Kundu, Z. Lu, V. Rathod, T. Funkhouser, C. Pantofaru, D. Ross, L. S. Davis, and A. Fathi, "Dops: Learning to detect 3d objects and predict their 3d shapes," 2020.

[22] J. Ye, Y. Chen, N. Wang, and X. Wang, "Online adaptation for implicit object tracking and shape reconstruction in the wild," 2022.
[23] Y. Liu, K. Zhu, G. Wu, Y. Ren, B. Liu, Y. Liu, and J. Shan, "Mv-deepsdf: Implicit modeling with multi-sweep point clouds for 3d vehicle reconstruction in autonomous driving," 2023.
[24] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12697–12705, 2019.
[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.
[26] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.
[27] U. von Luxburg, "A tutorial on spectral clustering," 2007.
[28] M. Liu, E. Yurtsever, X. Zhou, J. Fossaert, Y. Cui, B. L. Zagar, and A. C. Knoll, "A survey on autonomous driving datasets: Data statistic, annotation, and outlook," *arXiv preprint arXiv:2401.01454*, 2024.
[29] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," *Applied Soft Computing*, vol. 70, pp. 41–65, 2018.