3D LIDAR SEGMENTATION BASED ON EUCLIDEAN CLUSTERING FOR EMBEDDED SYSTEM

Minh-Hai Le¹, Song-Toan Tran²

Abstract – This paper presents a method for 3D LiDAR segmentation based on Euclidean clustering specifically designed for embedded systems. LiDAR sensors are widely used for perception tasks in autonomous vehicles, robotics, and other applications, providing dense point cloud data of the environment. Segmentation of the point cloud into meaningful objects is essential for understanding the surroundings and making informed decisions. Euclidean clustering stands as an efficient technique, grouping points according to their spatial closeness and enabling object segmentation. However, deploying such algorithms on embedded systems comes with challenges due to limited computational resources. This study introduces a refined adaptation of the Euclidean clustering algorithm, specifically tailored for embedded systems. The study aims to ensure real-time performance within such constrained environments. The proposed approach involves acquiring raw point cloud data from the LiDAR sensor and preprocessing it to reduce noise and size. Adaptive Euclidean clustering is then applied to group points into clusters based on their spatial proximity. Extracted features such as centroids and bounding boxes are utilized for object classification and segmentation. Post-processing steps refine the segmentation results, improving accuracy and removing spurious clusters.

Keywords: 3D LiDAR, embedded system, Euclidean clustering, object detection, real-time performance, segmentation.

1-2 Tra Vinh University, Vietnam
*Corresponding author: lmhai@tvu.edu.vn
Received date: 06th July 2023; Revised date: 28th August 2023; Accepted date: 29th August 2023

I. INTRODUCTION

Three-dimensional light detection and ranging (3D LiDAR) sensors have become vital components in various fields, including autonomous vehicles, robotics, and environmental monitoring [1–3]. These sensors provide high-resolution point cloud data, enabling accurate perception and understanding of the surrounding environment. Segmentation of the point cloud into meaningful objects is a fundamental task for many applications, as it allows for object detection, tracking, and scene understanding. 3D object detection aims to recognize the target of the movement by using the obtained 3D points cloud [4]. So, this technology has been widely used in autonomous driving, robot technology, pattern recognition, virtual reality, and other fields [5, 6].

In the field of autonomous vehicles, 3D object detection under harsh weather conditions is one of the most important steps. Figure 1 depicts the LiDAR data acquired during adverse driving conditions characterized by heavy snowfall, sourced from the Canadian Adverse Driving Conditions Dataset [7]. It is discernible that the efficacy of LiDAR monitoring experiences a notable diminishment amid snowy conditions. The blue circle demarcated in Figure 1 encapsulates spurious LiDAR points that are an outcome of snowflakes. The advent of snow within the point cloud transpires haphazardly, consequently obstructing the optical clarity crucial for vehicular navigation. Notably, the density of snow manifestations within the point cloud is distinctly concentrated within a range of 20 meters, progressively waning with distance. This intricate phenomenon emerges as a consequential predicament, engendering substantial concerns for potential vehicular accidents, particularly within the domain of autonomous vehicles.
Euclidean clustering is a widely used method for segmenting 3D LiDAR point clouds [8]. It groups points together based on their spatial proximity, assuming that points belonging to the same object are close to each other. Euclidean clustering has proven to be effective in segmenting objects such as cars, pedestrians, and buildings. However, implementing this algorithm on resource-constrained embedded systems poses several challenges. The segmentation of sensory data into discernible objects remains a formidable challenge within the realm of computer vision. Numerous methodologies have been employed to segment point cloud data, with clustering being the predominant approach. The utilization of clustering methodologies is widespread due to its inherent simplicity and ease of implementation, rendering it an indispensable tool in the segmentation of point cloud data. Nevertheless, conventional clustering approaches are characterized by sluggishness and instability, particularly when confronted with disparate data distributions across varying distances.

In pursuit of an optimal balance between efficiency and precision, this study undertakes a multi-pronged approach to enhance the existing algorithm. Primarily, the implementation of a k-d tree data structure facilitates the establishment of an index structure within the point cloud data, thereby substantially amplifying retrieval speeds. Subsequently, an adaptive filter is harnessed to selectively attenuate noise points while conserving the intricate details inherent to the object point cloud data. The pivotal task of distinguishing ground from objects is aptly addressed through the adoption of the random sample consensus (RANSAC) technique. The pinnacle of this proposed algorithm lies in the employment of adaptive Euclidean clustering, wherein distinct thresholds are adeptly tailored to accommodate escalating distances. This paper proposes a method for 3D LiDAR segmentation based on Euclidean clustering tailored for embedded systems. The primary objective is to address the challenges of limited resources while maintaining reliable object segmentation in real-time. The proposed method takes into account the computational constraints of embedded systems and presents optimizations to enhance the efficiency of the algorithm.

The paper is structured as follows: section 2 provides an overview of related work in the field of 3D LiDAR segmentation; section 3 describes the proposed approach, including the preprocessing steps, the adaptive Euclidean clustering algorithm, and the post-processing techniques employed; section 4 presents experimental results and performance evaluation on a representative embedded system; and section 5 concludes the paper with a summary of the contributions and discusses potential future directions for research in this area.

Fig. 1: LiDAR data under snow condition

Overall, this paper aims to provide a practical and efficient solution for 3D LiDAR segmentation based on Euclidean clustering, tailored specifically for embedded systems. By addressing the limitations of computational resources, the proposed method enables real-time object segmentation, opening up possibilities for a wide range of applications in autonomous navigation, robotics, and beyond.
II. LITERATURE REVIEW

Segmentation of 3D LiDAR point clouds using Euclidean clustering has been a topic of extensive research in the field of computer vision and robotics. Several studies have proposed different algorithms and techniques to improve the accuracy and efficiency of object segmentation. However, implementing these algorithms on embedded systems with limited resources poses additional challenges. This section provides an overview of some relevant works in the area of 3D LiDAR segmentation based on Euclidean clustering, focusing on the efforts made to address the requirements of embedded systems.

One of the fundamental contributions in this area introduced the concept of Euclidean clustering for LiDAR point cloud segmentation [9]. Their approach demonstrated the effectiveness of this approach in segmenting objects in urban environments. However, their work primarily focused on desktop computing platforms and did not consider the constraints of embedded systems.

To address the challenges of limited resources in embedded systems, a parallel implementation of the Euclidean clustering algorithm using graphics processing units (GPUs) was proposed [10]. By leveraging the parallel computing capabilities of GPUs, they achieved significant speedup in the segmentation process. While this work demonstrated the potential of using parallel processing for improving performance, it did not specifically optimize the algorithm for embedded systems.

A hierarchical clustering approach for 3D LiDAR segmentation was proposed to tackle the efficiency requirements of embedded systems [11]. Their method efficiently reduced the search space by partitioning the point cloud into smaller regions of interest. By iteratively applying Euclidean clustering at diverse levels of the hierarchy, they achieved improved segmentation performance with reduced computational complexity. However, the implementation details for embedded systems were not explored in this work. In recent years, there have been efforts to develop specialized hardware architectures for LiDAR point cloud processing. A hardware accelerator for real-time 3D LiDAR segmentation was introduced. They designed a custom architecture that leveraged parallelism and optimized memory access patterns to achieve high throughput while meeting the resource constraints of embedded systems.

In addition to algorithmic and hardware optimizations, some studies have explored the use of machine learning techniques to enhance the accuracy of 3D LiDAR segmentation. For instance, combining Euclidean clustering with a deep learning framework to improve the segmentation results has also been interesting in recent years. Their approach incorporated semantic information from the clustered points to refine the object boundaries and achieve more precise segmentation.

While these works have made significant contributions to the field of 3D LiDAR segmentation, there is still a need for dedicated research on optimizing the Euclidean clustering algorithm specifically for embedded systems. This paper aims to address this gap by proposing an optimized approach that considers the constraints of embedded systems, allowing for real-time 3D LiDAR segmentation with reliable accuracy.

III. RESEARCH METHODS

The proposed approach in ‘3D LiDAR Segmentation based on Euclidean Clustering for Embedded System’ aims to optimize the segmentation process of 3D LiDAR point clouds specifically for embedded systems. The approach considers the limited computational resources of embedded systems, such as processing power, memory, and energy consumption, while ensuring real-time performance and reliable object segmentation. The proposed approach follows a systematic pipeline consisting of several key steps as Figure 2.

Data Acquisition: The raw point cloud data is obtained from the LiDAR sensor, which provides 3D coordinates (x, y, z) for each point in the environment.
Preprocessing: To reduce noise and minimize the computational load, appropriate preprocessing techniques are applied to the point cloud data. The proposed system uses ground removal and adaptive filter to obtain a clean and compact representation of the point cloud.

Adaptive Euclidean clustering: The core of the approach involves applying the Euclidean clustering algorithm to group points in the preprocessed point cloud based on their spatial proximity. Points within a dynamic distance threshold are considered part of the same cluster, representing a distinct object. Euclidean clustering effectively segments the point cloud into clusters corresponding to different objects in the scene.

Segmentation: Relevant features are extracted from each cluster to provide additional information about the objects. Based on the extracted features, the clusters are classified into different object categories or assigned unique labels.

Embedded system optimization: To adapt the algorithm for embedded systems, various optimization strategies are employed. These strategies take into account the limited computational resources of the target hardware. Techniques such as parallel processing, optimization algorithms, and efficient data structures are utilized to ensure real-time performance while minimizing resource usage.

The proposed approach is designed to be implemented on embedded systems, enabling efficient and real-time 3D LiDAR segmentation. Optimizing the algorithm specifically for embedded environments provides a practical solution for applications requiring accurate object segmentation, such as autonomous navigation, robotics, and environmental monitoring.

A. Ground Segmentation Based on RANSAC

Ground segmentation based on RANSAC (Random Sample Consensus) is a popular technique used in point cloud processing to separate ground points from non-ground points [14]. RANSAC is a robust algorithm that iteratively fits a mathematical model to a subset of the data, filtering out outliers and estimating the parameters of the underlying ground plane. The steps are as follows:

1. Random Sample Selection: Randomly select a subset of points from the point cloud dataset. The number of points selected should be sufficient to estimate the ground plane’s parameters accurately. Typically, three points are chosen to define a plane in 3D space.

2. Model Fitting: Use the selected points to fit a mathematical model that represents the ground plane. In the case of ground segmentation, the model is usually a 3D plane equation (ax + by + cz + d = 0). This equation represents the ground plane’s geometry, where the parameters (a, b, c, d) need to be estimated.

3. Inlier Classification: Classify the remaining points in the dataset based on their proximity to the estimated ground plane. Points that lie within a certain distance threshold from plane S are considered inliers and likely belong to the ground surface. Points outside the threshold are considered outliers and are likely part of non-ground objects.

4. RANSAC Iterations: Repeat steps 1-3 N times, and select the highest score of the plane S.

The RANSAC-based ground segmentation approach is effective in handling uneven terrain, varying ground surfaces, and complex point cloud data. It is robust to outliers and noise, as the algorithm focuses on finding a consistent set of inliers to estimate the ground plane parameters. By separating ground points from non-ground points, this technique is valuable for various applications, including terrain mapping, autonomous navigation, and environmental analysis.
B. The adaptive euclidean clustering

This research considers a scenario where m data points are initially divided into n classes. To establish the relationship between these points, a measure called the similarity of distance is defined. The proposed approach involves iteratively combining the two classes with the smallest distance into a single class. This process calculates the distance between classes, until either the distance between all categories exceeds a fixed threshold or the number of classes reduces to a specified number, indicating completion of the split.

To differentiate between different classes, the Euclidean distance is utilized in this study. The Euclidean distance between two points, denoted as P and Q, corresponds to the length of the line segment connecting them. In Cartesian coordinates, given two points p = (p1, p2, ..., pn) and q = (q1, q2, ..., qn) in Euclidean n-space, the distance from point p to q can be computed using the Pythagorean formula:

\[ d(p, q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \]

Segmentation based on Euclidean distance relies on evaluating the distances between neighboring points. The key parameter here is the radius threshold. In contrast to the conventional DBSCAN approach [15], the adaptive Euclidean clustering dynamically fine-tunes the density threshold during the clustering process. This adaptive adjustment guarantees accurate segmentation of clusters with diverse densities, allowing for an effective treatment of intricate formations and changing densities inherent in LiDAR point clouds.

Adaptive Euclidean clustering has exhibited promising outcomes in the segmentation of LiDAR, the distribution of point clouds is not uniform as the distance from the sensor increases. This phenomenon is clearly illustrated in Figure 3. Consequently, utilizing a fixed parameter to process data across varying distances might yield suboptimal results. To enhance the efficacy of clustering, the study employs distinct radius thresholds corresponding to different distance regions. The parameter "eps" signifies the minimum resolution between two points at the farthest range measured by the LiDAR sensor. It can be calculated using the following equation:

\[ \text{eps} = \alpha \times \theta \times R_{a,b} \]

where \( R_{a,b} \) denotes the distance from the LiDAR to objects. The variable “\( \alpha \)" stands for the multiplication factor, and “\( \theta \)” represents the angular resolution (azimuth) of the LiDAR. The entire process of the segmentation methodology is visually presented in Algorithm 1.

Algorithm 1. Pseudocode of the proposed method

Def adaptiveCluster:
C=0
for p ∈ P do
if p was not labeled:
dist=√(x²+y²+z²)
deck=KDTree search(SR)
F=count the number of neighbors N.
if N < MinPts:
P ← noise point (outlier)
else
C=C+1
expanderCluster (P,F,C,SR,MinPts)
end if
end if
end for.
Def expandCluster(P,NbPts, C,eps, MinPts):
add P to cluster C
for each point P in NbPts do:
if P is not visited:
mark P as visited.
NbPts = regionQuery (P, eps)
if sizeOf (NbPts) ≥ MinPts:
NbPts = NbPts joined with NbPts'
end if
end if
if P is not yet member of any cluster:
add P to cluster C.
end if
end for.

95
IV. RESULTS AND DISCUSSION

In this section, experiments using conditions mimicking embedded board scenarios were performed to assess the efficacy of the proposed segmentation method in challenging weather conditions. The outcomes derived from the implementation yielded promising results. Notably, the segmentation algorithm demonstrated a commendable ability to accurately differentiate and segment distinct objects within the point cloud data.

A. Implementation results

The study assessed the performance of the proposed method using unprocessed data obtained from the WADS dataset. This dataset encompasses a wide range of scenarios, spanning various weather conditions from mild illumination to severe snowfall. The outcomes of applying the proposed method are visually presented in Figure 4. The initial set of images in the figure depicts the unprocessed point cloud data extracted from the WADS dataset. The subsequent set of images demonstrates the outcomes of the ground removal process, and the final image unveils the results achieved after the implementation of the segmentation algorithm. Within Figure 4 a, a notable observation is the presence of numerous densely packed ground points near the vehicle. Post the filtration process, depicted in Figure 4 b, there is a discernible absence of ground and snow particles, while crucial objects like the car and the tree are effectively retained. The method highlights a high level of accuracy when compared with the ground truth. Notably, a significant number of ground particles were successfully eliminated from the data derived from the road surface.

The comparison of the approach with the DBSCAN method is illustrated in Figure 4 c. The DBSCAN effectively segments objects within the LiDAR point cloud. However, a drawback is that it removes object points as well. Moreover, DBSCAN struggles with object points situated at greater distances, which can negatively impact object recognition. The proposed method, on the other hand, retains more object points at far distances compared to DBSCAN. Interestingly, it proposed filter successfully classifies two nearby cars. The proposed filter not only optimizes point object detection but also efficiently eliminates noise from snow particles. As shown, the method generates a significant number of clustering results, effectively restoring the real scene, as depicted in Figure 4 d. In contrast, Figure 4 c exhibits a loss of detail due to the fixed radius value. The undersized radius in Figure 4 c leads to points within classes falling below the threshold. Based on the experiments, the utilization of a variable radius aligns better with real-world scenarios. This approach ensures that the segmentation remains appropriate across varying circumstances.
Fig. 4: Comparison between the state before and after the application of the proposed method

B. Evaluation

In this section, the filter performance will be evaluated by comparing processing time and accuracy with the DBSCAN segmentation.

Table 1: Comparison of the proposed segmentation

<table>
<thead>
<tr>
<th></th>
<th>The proposed method</th>
<th>DBSCAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numbers of cluster</td>
<td>222</td>
<td>161</td>
</tr>
<tr>
<td>False negative rate (%)</td>
<td>3.8</td>
<td>5.25</td>
</tr>
<tr>
<td>Frame per second</td>
<td>0.3</td>
<td>0.16</td>
</tr>
</tbody>
</table>

To evaluate segment efficiency, the study employs labels for the car point clouds from the WADS dataset. These labels encompass 20 scenes, each containing over 100,000 points in every frame. This dataset serves as a basis for comparing the False Negative rate with the DBSCAN method. Subsequently, the number of clusters and processing time are computed using the labeled dataset. The segmentation speed of each method is tabulated in Table 1, where the proposed approach exhibits a processing speed approximately twice as fast as DBSCAN. This performance is substantiated by Table 1’s comparative analysis of the method against DBSCAN segmentation.

The proposed method effectively tackles the complexities posed by fluctuating point densities and non-uniform distributions often encountered in LiDAR point clouds. At its core, the Adaptive Density-Based Spatial Clustering (ADBSAC) method hinges on the notion of dynamically adjusting the density threshold inherent to DBSCAN. This adaptability facilitates a finer degree of segmentation precision, particularly within areas characterized by varying point densities.

V. CONCLUSION AND RECOMMENDATIONS

The outcomes of the proposed method offer illuminating insights into its remarkable performance. Managing effectiveness in extreme environments has remained a challenging endeavor in recent research undertakings. Experiments conducted with the WADS dataset provide robust confirmation of the algorithm’s prowess in curbing the presence of unwanted snow-related points during data collection. This study underscores the noteworthy utility and efficacy of the proposed method, which stands as a vital synthesis of RANSAC and adaptive Euclidean clustering. Notably, there are instances where the True Positive (TP) rate marginally trails behind recent filters. However, post the application of the proposed filter, the False Negative (FN) rate demonstrates a consistent diminishment across diverse weather conditions. The resulting point clouds post-segmentation exhibit remarkable clarity.

Evidenced by the FN rate below 5%, the proposed method achieves commendable performance in numerous scenarios, particularly in the
context of vehicle driver assistance systems. The integration of RANSAC and ADBSAC augments LiDAR efficiency and bolsters its operational capability within challenging environmental contexts. This amalgamation not only holds potential for refining noise filtration and point cloud segmentation but also charts an innovative trajectory in pre-processing for machine learning applications.

REFERENCES


