

## Research Article

# A Novel Method for 3D Object Detection in Open-Pit Mine Based on Hybrid Solid-State LiDAR Point Cloud

Cheng Li,<sup>1,2</sup> Gang Yao ,<sup>2</sup> Teng Long ,<sup>2</sup> Xiwen Yuan ,<sup>2</sup> and Peijie Li<sup>2</sup>

<sup>1</sup>Institute of Railway and Urban Rail Transit of Tongji University, Shanghai 201804, China

<sup>2</sup>CRRCC Zhuzhou Institute Co., Ltd., Zhuzhou, Hunan 412001, China

Correspondence should be addressed to Gang Yao; [yogi@hnu.edu.cn](mailto:yogi@hnu.edu.cn)

Received 4 October 2023; Revised 29 November 2023; Accepted 25 January 2024; Published 14 February 2024

Academic Editor: Sivakumar Poruran

Copyright © 2024 Cheng Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, the mining industry has encountered challenges, such as a shortage of human resources, an ongoing emphasis on safety enhancements, and increased ecological preservation requirements. Autonomous mining trucks have emerged as a novel solution to effectively address these issues within open-pit mining operations. To meet the demanding conditions of open-pit mines, characterized by intense vibrations and extreme temperature variations, hybrid solid-state LiDAR has emerged as the primary choice for perception sensors. Recognizing the distinct data structure and distribution disparities between point clouds obtained through nonrepetitive scanning methods of hybrid solid-state LiDAR and traditional mechanical LiDAR, this paper proposed an innovative LiDAR 3D object detection model, PointPillars-HSL (PointPillars-Hybrid Solid-state LiDAR). This approach harmonizes the unique characteristics of open-pit mining environments and hybrid solid-state LiDAR point clouds. It optimizes the model's preprocessing methodology, augments the dimensionality of pillar features, fine-tunes the loss function, and employs transfer learning techniques to reduce the reliance on specific datasets. The result is the effective deployment of a 3D object detection algorithm customized for hybrid solid-state LiDAR within the specific operational framework of open-pit mining. This achievement has yielded a noteworthy overall vehicle recognition rate of 89.72%.

## 1. Introduction

**1.1. Application Background.** In recent years, the autonomous driving industry has prospered, but there are very few autonomous driving projects that have actually achieved actual benefits. For short-distance, fixed-site, high-intensity open-pit mine transportation, autonomous driving technology has high practical value, and closed mining areas are also one of the most likely scenarios for autonomous driving. An automatic driving system of a mining truck generally consists of an autonomous mining truck, a communication system, and a central monitoring system, as shown in Figure 1. These systems work together to achieve efficient scheduling and operation in open-pit mining. The advantages of autonomous mining trucks are reflected in not only solving the transportation problems in smart mines and improving the efficiency of transportation operations in open-pit mines but also reducing safety accidents in open-pit mines, optimizing production management, solving the problem of difficult

recruitment of mining truck drivers, and reducing labor costs.

The transportation road in the open-pit mine is an unstructured road without obvious road structure characteristics, and the road surface is relatively undulating. A large amount of dust in dry weather and accumulated water and snow after rain and snow will also have a great impact on road conditions. In such a complex and changeable environment and nighttime operating conditions without external lighting, camera-based perception cannot stably obtain surrounding environment information. The LiDAR-based perception method is more adaptable to the mining environment and can meet the 7 × 24 hr of autonomous mining truck operation requirements.

**1.2. Hybrid Solid-State LiDAR and Point Cloud.** Although LiDAR is the best perception sensor for autonomous mining trucks, the production conditions in the open-pit mine are very harsh. The LiDAR mounted on the mining truck not

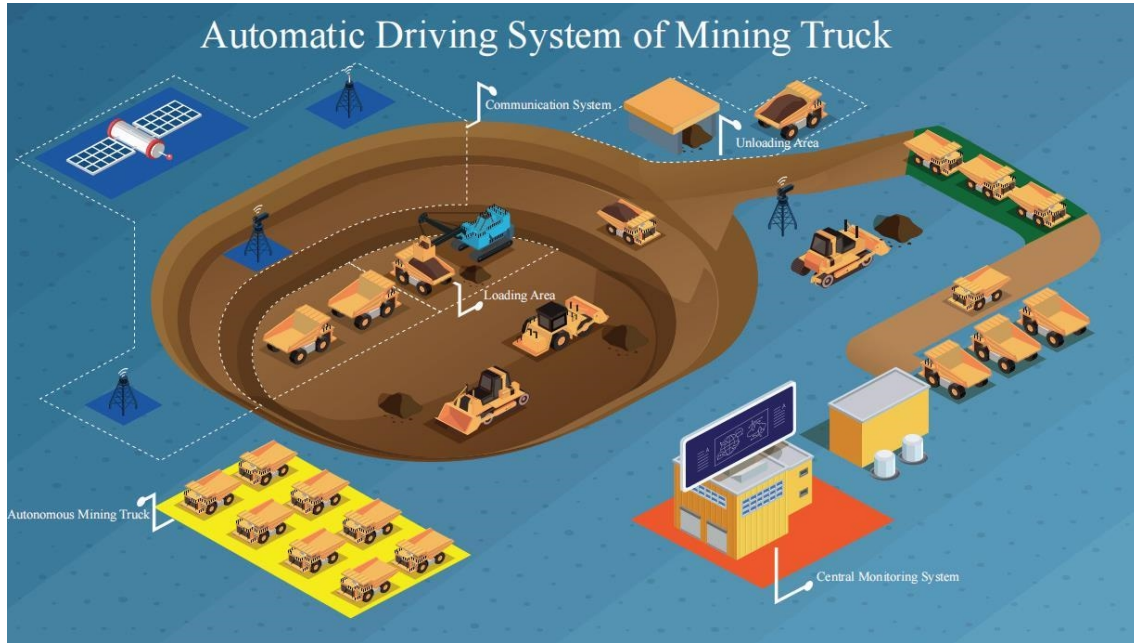


FIGURE 1: Automatic driving system of mining truck.

only has to withstand the test of high and low temperatures and strong vibration but also faces severe mechanical shock during loading and unloading. Therefore, the performance of LiDAR directly determines the accuracy and stability of environmental perception.

LiDAR mainly includes laser emission, reception, scanner, and signal processing circuits. According to different scanning methods, LiDAR can be divided into three types: mechanical LiDAR, hybrid solid-state LiDAR (semi-solid-state LiDAR), and solid-state LiDAR. The mechanical LiDAR drives the optical-mechanical structure to rotate 360° through a motor and can scan the surrounding environment in all directions to form a point cloud. However, the mechanical LiDAR adopts the traditional discrete design, which is bulky and relies heavily on moving parts, so it is not suitable for application in the environment of extreme temperature variations and strong vibration and shock in open-pit mines. Compared with mechanical LiDAR, hybrid solid-state and solid-state LiDAR generally can only achieve a horizontal field of view of 120°, but hybrid solid-state has fewer rotatable moving parts, and solid-state has no mechanical movement, which has higher stability and is shrinking in size and cost. The hybrid solid-state LiDAR is currently the popular LiDAR solution for mass-produced vehicles, and its technologies are relatively mature.

The data collected by LiDAR is called point cloud. A point cloud is a collection of massive points that express the spatial distribution of targets and surface characteristics of objects in a 3D space coordinate system. The content of the point cloud includes 3D coordinates ( $XYZ$ ) and reflection intensity (Intensity). The intensity information is related to the surface material, roughness, incident angle and direction of the object, as well as the emitted energy of the laser and the wavelength of the laser.

The point cloud itself has a natural disorder, which means that exchanging any point will not affect the 3D description of the spatial object. Similarly, translation, rotation, and scaling will not affect its 3D description. This feature can be used to perform data augmentation on point clouds before network training. In addition, point clouds are also characterized by sparsity, which has low resolution in 3D space, so 3D convolutions [1] performed on point clouds are not as effective as 2D convolutions on images.

The characteristics of the open-pit mine's lidar point cloud lie in the significant differences in the distribution of the  $z$ -coordinates of object centers beneath undulating roads, with object sizes generally larger than those in urban scenes. The environment is characterized by unstructured roads, and dust is commonly present. The dust not only introduces noise to the point cloud data but also accumulates on the specular surfaces of the lidar, leading to an overall decrease in the quality of the point cloud data. Additionally, the vibration during the movement of vehicles in the mining area is much greater than that in an urban scene, which can interfere with point cloud data, resulting in data distortion.

*1.3. LiDAR Perception Algorithm.* Open-pit mines generally have strict restrictions on the entry and exit of vehicles and pedestrians, so the target obstacles appearing in the mining stripping section of the open-pit mine are mainly vehicle types, including electric shovels, mining trucks, bulldozers, road graders, sprinklers, etc., and the vehicle is large and easy to detect, and the probability of pedestrians is extremely low, which is more suitable for the application of object detection methods in deep learning.

3D object detection methods in deep learning rely on data-driven [2] input of massive sample data, and the detection accuracy can exceed the traditional method of feature

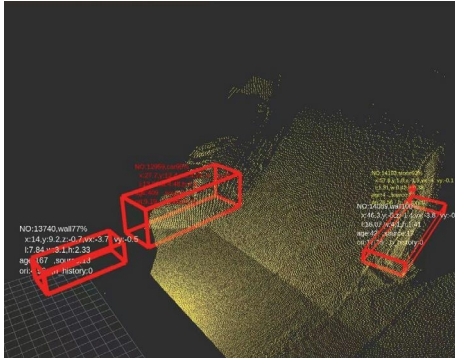


FIGURE 2: False detection of vehicle (taken from a continuous frame).

extraction. Therefore, data become the key to deep learning algorithms. At present, the public datasets of autonomous driving containing point cloud data are increasing year by year and cover tasks such as object detection, object tracking, semantic segmentation, mapping, and positioning, such as KITTI [3], SemanticKITTI [4], BLVD [5], Waymo, nuScenes, and so on. However, the cover scenes of these datasets are mainly on structured roads in cities, while open pit mines are unstructured scenes. The object types, object sizes, and perceived environments of the two scenarios are quite different, and the existing public datasets cannot be directly used as training data for 3D object detection on unstructured roads in open-pit mines. At present, the road conditions of open-pit mines are complex, and with the advancement of mining, the road environment will always be changed, which brings great challenges to data collection. In addition, in the current open-pit mine dataset, the proportion of mining trucks in the dataset is very high, reaching more than 50%, while the proportion of other objects is relatively small, resulting in poor class balance in the dataset.

The traditional LiDAR point cloud processing method typically begins by segmenting and extracting nonground points, then clusters obstacle points to obtain obstacle features, and performs multiobject matching and tracking based on object features. The complex and changeable road conditions of open-pit mines bring challenges to the nonground point segmentation and extraction of traditional point cloud detection methods. Moreover, the traditional method cannot distinguish the types of vehicles in open-pit mines, and the generalization ability of point cloud features based on rule extraction is limited. For problems such as false positives of vehicles caused by street signs and retaining walls, as shown in Figure 2, traditional point cloud processing methods may not be as good as deep learning methods.

Some early algorithms based on deep learning, such as point-based PointNet [6] and PointNet++ [7], these type of algorithms need to map the point set features back to the original point cloud after calculating the neighborhood features, which have a large time complexity. The voxel-based algorithm represented by VoxelNet [8] has a small proportion of nonempty voxels, and the data expression is inefficient. Moreover, the calculation of the 3D convolution calculation features used is very computationally intensive,

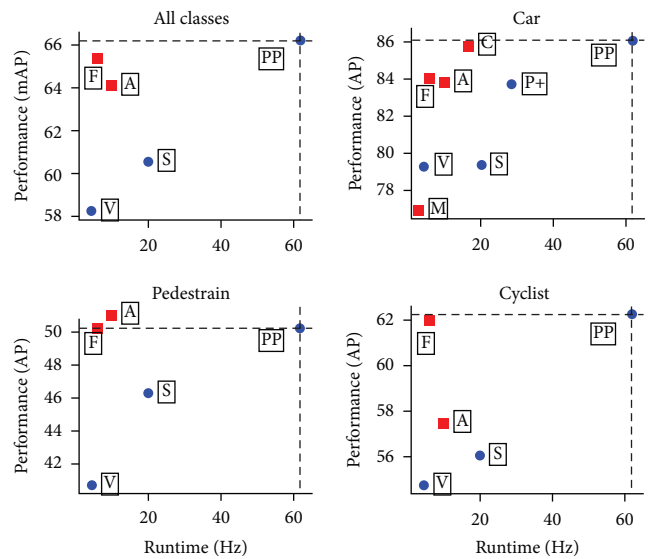


FIGURE 3: Performance comparison between PointPillars algorithm and other algorithms based on BEV perspective.

and the reasoning is very time-consuming. Some methods, such as AVOD [9], project point clouds onto a 2D plane and employ image-based approaches for object detection. However, the projection process inevitably leads to the loss of certain geometric spatial information, resulting in shortcomings in depth prediction. In this context, PointPillars [10, 11] has garnered widespread attention due to its ability to strike a favorable balance between inference speed and detection accuracy. PointPillars achieves efficient object detection by converting point cloud data into a compact voxel representation and employing a columnar structure-based processing approach. Compared to other Bird's Eye View (BEV)-based methods, PointPillars demonstrates a significant advantage by consistently improving inference speed to 60 frames per second (60 Hz) while maintaining a certain level of detection accuracy. Figure 3 illustrates a comparison between PointPillars and other BEV-based methods, such as Frustum PointNet [12] and PIXOR++ [13] in terms of detection accuracy and speed. It is evident that PointPillars manages to achieve higher inference speed while maintaining a certain level of detection accuracy. Baidu Apollo 6.0 has pioneered the adoption of the PointPillars-based algorithm for LiDAR point cloud detection, achieving a threefold increase in detection frequency compared to SECOND [14].

However, although the object features extracted by PointPillars include the relative distances of each point to the arithmetic mean of all points within the pillar, the subsequent pooling operations do not effectively preserve this feature. A major feature of the open-pit mine dataset is that the obstacles are concentrated as vehicles, and the target size is large. The pillar that contains the point cloud of the vehicle is quite different from other Pillars that do not contain obstacles in point cloud density and point cloud distribution in the  $z$ -direction.

In light of the challenges posed by the complex and dynamic nature of unstructured road environments in

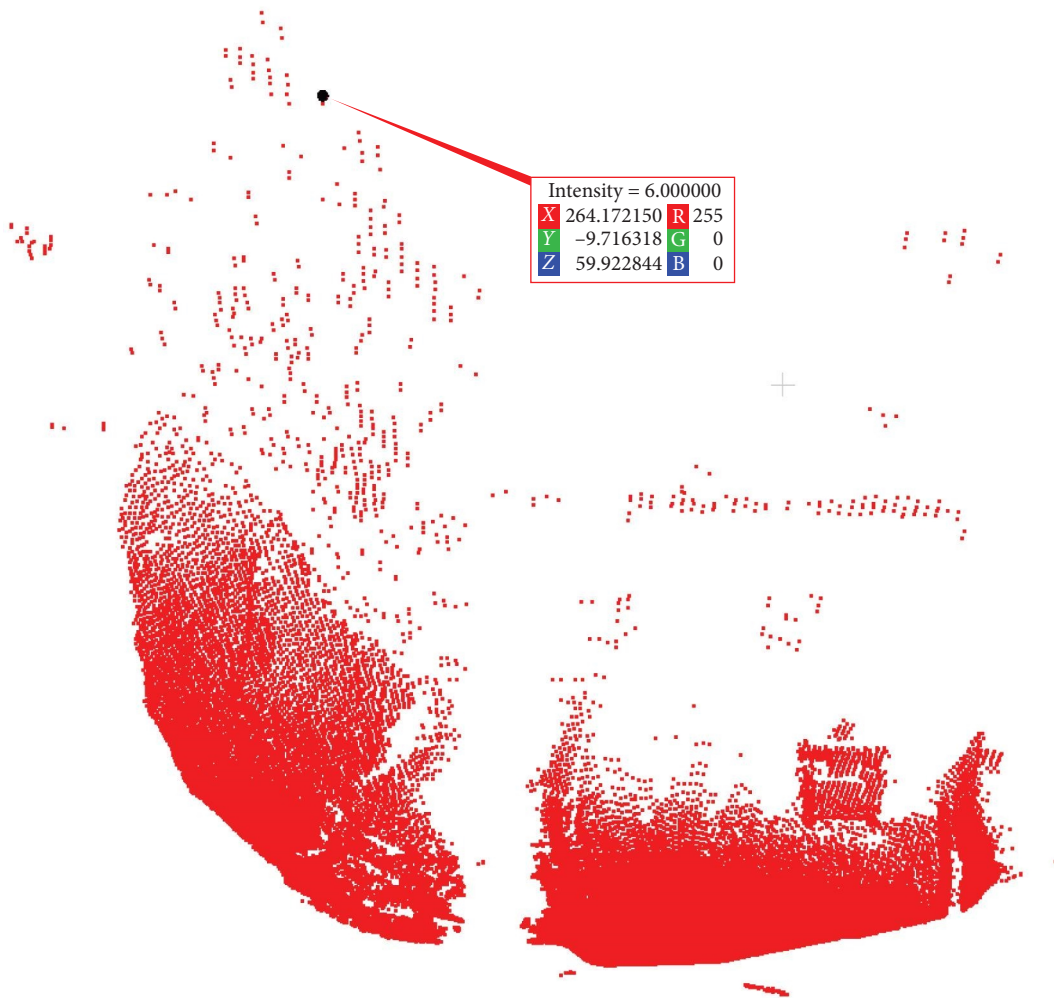


FIGURE 4: Point cloud of innovusion.

open-pit mining, as well as the harsh operating conditions and lack of suitable datasets, this paper presents a novel 3D object detection model for LiDAR, named PointPillars-HSL (PointPillars-Hybrid Solid-State LiDAR). This model is designed by taking into account the characteristics of hybrid solid-state LiDAR point clouds in practical applications. This approach builds upon the foundation of the PointPillars model by optimizing the preprocessing structure and loss function, enhancing the dimensions of point cloud features, and utilizing transfer learning [15] methods to mitigate the need for specific datasets. As a result, improved detection performance is achieved on both the training and deployment platforms.

## 2. PointPillars-HSL Object Detection Algorithm

*2.1. Data Preprocessing.* The three elements of artificial intelligence are data, algorithms, and computing power, and data plays a leading role in the three elements. The detection dataset of the open-pit mine comes from the LiDAR of Innovusion mounted on the mine truck and Figure 4 shows a

frame of mining area point cloud data collected by Innovusion. The LiDAR has a maximum detection range of 500 m, with a detection range of 250 m at 10% reflectivity. It features a 120° horizontal field of view, a 25° vertical field of view, and a resolution of  $0.18^\circ \times 0.24^\circ$ . The ranging accuracy achieves  $\pm 5$  cm.

The PointPillars algorithm filters out point clouds that are projected outside the image bounds based on their projection in the image. However, due to distortion and instability in the fisheye cameras on the mining trucks, the PointPillars-HSL algorithm cannot perform filtering based on images. During the data preprocessing, an appropriate region of interest for point clouds is defined, allowing for initial downsampling of the point cloud data.

Due to the point clouds captured by the fixed-position LiDAR on mining trucks being influenced by the uneven terrain and steep road conditions in the open-pit mine, there is a wide distribution range of the Z-coordinates of the detected objects. If a small point cloud filtering range is applied in the z-direction, it will result in the exclusion of many obstacle targets. This article analyzes and counts the

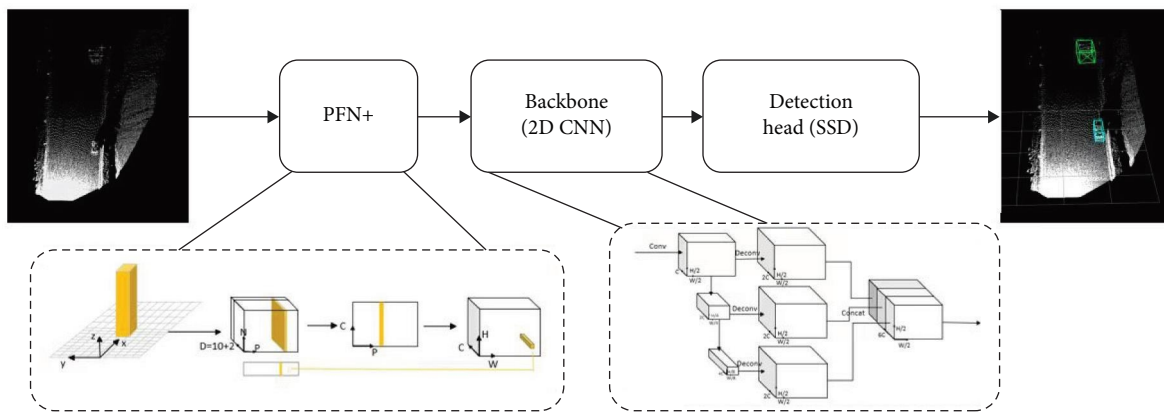


FIGURE 5: PointPillars-HSL network structure.

object height of the mining truck dataset to obtain the point cloud height range of the vehicle and dynamically adjusts it in the data preprocessing step of the model training process.

Compared with the anchor-free algorithm [16], the PointPillars-HSL algorithm is particularly sensitive to the setting of anchor boxes. It is difficult for the detection network to learn the object bounding box directly from the data. The role of the anchor is equivalent to giving the network a learning template about the object size in advance. Building upon this foundation, the network undergoes further learning to accurately derive 3D boundingbox boxes. The anchor parameters in the general PointPillars model are representative of vehicles in urban road scenarios and may differ from the vehicles encountered in open-pit mines. Consequently, after performing precise point cloud annotation on a subset, anchor box dimensions for different vehicle categories in the open-pit mine were determined.

**2.2. Transfer Learning.** Open-pit mines have a limited number of vehicles, and the close relationship between vehicle categories and operational scenarios leads to a relatively low efficiency in data collection. Additionally, the steep and unstructured terrain in open-pit mines introduces challenges in annotation. At present, the number of dataset samples in open-pit mines is relatively limited. If training is started from scratch, the stability and reliability of the model cannot be guaranteed. Therefore, this paper uses the transfer learning method to reduce the number of training data. Transfer learning is a branch of machine learning that exploits the similarity between data or tasks to apply a model trained in an old domain to model initialization in a new domain. The transfer learning approach utilized in this study is based on parameter transfer within a supervised learning framework. Given that shallow networks possess generic features and strong transferability, the model parameters trained on the KITTI urban road dataset were employed in the training of a detection model tailored to open-pit mining scenarios. The pretrained model serves as the starting point

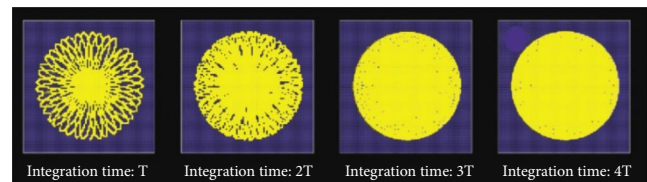


FIGURE 6: The coverage of point clouds from nonrepetitive scanning varies over time.

for model learning, with further adjustments to the parameters of the deep network aimed at enhancing the model's generalization performance.

Transfer learning is founded on two fundamental concepts: domain and task. Domains are divided into source and target domains. In this paper, the source domain is the urban road scene, while the target domain is the open-pit mine. The task being learned in both domains is consistent, involving 3D object detection. Moreover, the designed model structures are highly similar, making them suitable for parameter-based transfer learning. Leveraging the abundant labeled samples and domain adaptation available in the urban road scene, transfer learning can alleviate the challenges associated with learning vehicle features in the open-pit mine.

**2.3. Model Structure.** The PointPillars-HSL network structure proposed in this paper is mainly divided into three parts: PFN+ (Pillar Feature Net+), 2D Backbone [9], and SSD detection head, as shown in Figure 5.

In the PFN+ module, the downsampling of point clouds is achieved by adjusting the division of Pillars (cuboidal point cloud segments). This approach effectively addresses the issue of nonuniform distribution of distant and nearby point clouds. Compared to the 360° rotating scan of mechanical LiDARs, the nonrepetitive scanning approach of hybrid solid-state LiDAR ensures that the scanning path does not repeat and the illuminated area within the field of view increases over time. Figure 6 illustrates the accumulation of

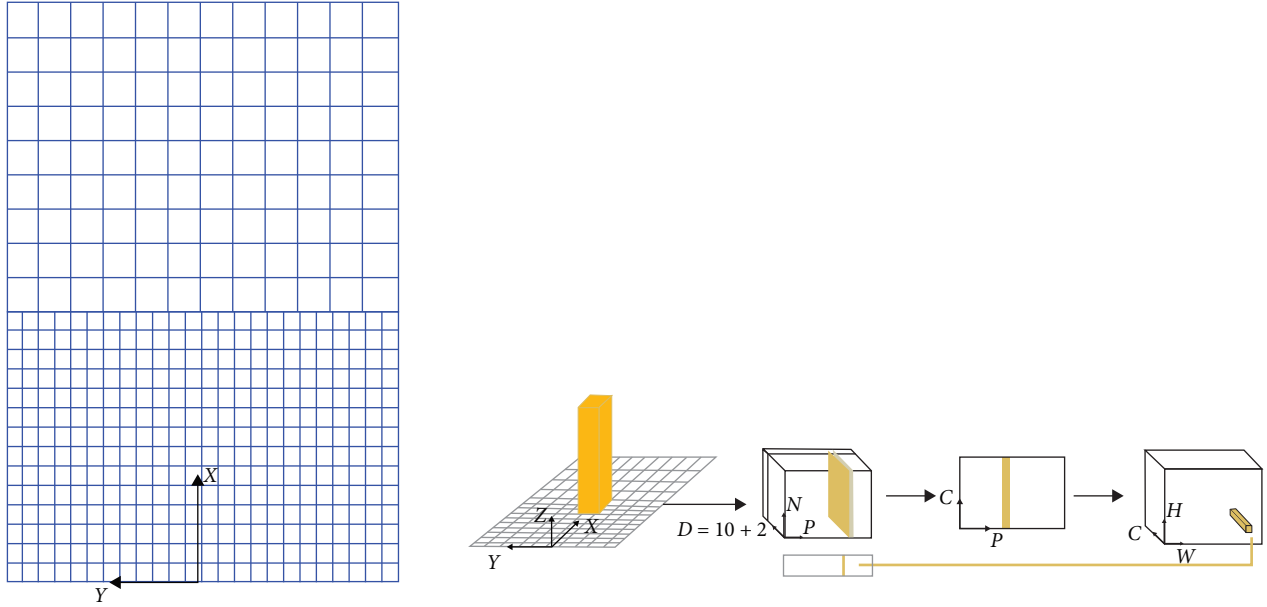


FIGURE 7: Pillar grid division and PFN+ module.

point clouds from the hybrid solid-state LiDAR as time progresses.

The hybrid solid-state LiDAR point clouds exhibit a distinct characteristic of being sparse in distance and dense in proximity. To optimize the pillar generation in the PointPillars-HSL model, a different approach is taken during the pillar generation phase. Instead of discretizing the point clouds into a grid with uniform spacing, the model adapts to the actual characteristics of the point clouds in the open-pit mine. In this paper, smaller pillar sizes are selected for denser point clouds in closer proximity, leading to the generation of more features and achieving finer localization. Sparse point clouds in the distance are mapped to larger pillars. This strategy not only reduces the number of pillars but also enhances the features of distant point clouds, thus improving the detection performance. The entire process and details can be seen in Figure 7.

Due to the presence of pooling operations causing sparse features in the PFN+ module, the proposed PointPillars-HSL approach introduces additional features at each individual point to retain the pillar density and point cloud's  $z$ -direction distribution information extracted from all points within a pillar. Specifically, the process involves concatenating the offsets of each point within every pillar to both the center point of the pillar and the arithmetic mean point of all points within the pillar, along with the  $x, y, z$  coordinates, intensity values, density features, and  $z$ -direction characteristics of each point. After partitioning the point cloud into Pillars, constraints are applied to the nonempty pillar count ( $P$ ) per sample and the point count ( $N$ ) per pillar. This is done to construct a tensor of size  $(D, P, N)$ , where  $D=12$ , representing the feature dimensionality of points within Pillars. This augmented feature set includes the original 10 dimensions and incorporates density features and distribution characteristics. Subsequently, the tensorized point cloud data undergoes processing and feature extraction to yield a

tensor of size  $(C, P, N)$ . Finally, a max-pooling operation is applied along the third channel, resulting in an output tensor of size  $(C, P)$ . This encoded feature is then redistributed back to the original pillar positions, forming a pseudo-image of dimensions  $(C, H, W)$ , where  $P = H \times W$ .

The density calculation formula for each pillar is as follows, where  $N$  represents the number of points within each Pillar:

$$P_{\text{density}} = \min\left(1, \frac{\lg(N+1)}{\lg(150)}\right). \quad (1)$$

The vertical distribution of the point cloud can be reflected by the variance of the  $z$  coordinates of all points within each Pillar:

$$S^2 = \frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N}. \quad (2)$$

The Backbone (2D convolution) consists of two components: one sub-network performs progressive downsampling on the pseudo-image to extract features at different scales, while another network performs upsampling on features extracted from top to bottom, resizing the feature maps to match the original pseudo-image size. This facilitates channel-wise concatenation at a consistent scale. The PointPillars-HSL algorithm continues to adopt the SSD detection head [17], predicting the categories, positions, and orientations of the objects.

**2.4. Loss Function and Model Training.** By analyzing the object categories within the training dataset, it is evident that the category of mining trucks has the highest occurrence probability, while categories like water trucks, graders, and bulldozers have lower proportions. This imbalance in class distribution might lead to a bias toward predicting mining

trucks in the final predictions, consequently impacting the overall detection accuracy. In order to address the problem of imbalanced object categories within the training dataset, the algorithm employs Focal Loss [18] in the loss function. This allows for assigning higher weights to challenging samples, which are those that are prone to being misclassified. The key concept of Focal Loss involves introducing a tuning factor (focusing parameter) to adjust the weights of samples. This tuning factor results in low-difficulty samples contributing less weight in the loss calculation, while high-difficulty samples contribute greater weight. By utilizing Focal Loss, the weight of easily classifiable samples can be reduced, allowing the model to focus more on learning from challenging samples. This approach helps mitigate the impact of class imbalance in object detection tasks. The Focal Loss is as follows:

$$\mathfrak{L}_{\text{cls}} = -\alpha_a(1 - p^a)^\gamma \log p^a. \quad (3)$$

In the equation,  $p^a$  represents the predicted probability by the model, and  $\gamma$  is the tuning factor used to control the weight of challenging samples. When  $\gamma$  is set to 0, Focal Loss becomes equivalent to the cross-entropy loss function. Increasing the value of  $\gamma$  can further enhance the weight of difficult samples.

The loss function includes the position regression loss function, classification loss function, and orientation regression loss function.

The position regression involves components  $(x, y, z, w, l, h, \theta)$ . Taking the center point coordinate  $x$  as an example,  $x^{\text{gt}}$  represents the true value of the center point  $x$  in the annotated target, while  $x^a$  represents the predicted  $x$  value of the 3D anchor, with  $d^a = \sqrt{(w^a)^2 + (l^a)^2}$ . The calculation of the components for the position regression loss is as follows:

$$\Delta x = \frac{x^{\text{gt}} - x^a}{d^a}, \quad \Delta y = \frac{y^{\text{gt}} - y^a}{d^a}, \quad \Delta z = \frac{z^{\text{gt}} - z^a}{h^a}, \quad (4)$$

$$\Delta w = \log \frac{w^{\text{gt}}}{w^a}, \quad \Delta l = \log \frac{l^{\text{gt}}}{l^a}, \quad \Delta h = \log \frac{h^{\text{gt}}}{h^a}, \quad \Delta \theta = \sin(\theta^{\text{gt}} - \theta^a), \quad (5)$$

$$\mathfrak{L}_{\text{loc}} = \sum_{b \in (x, y, z, w, l, h, \theta)} \text{SmoothL1}(\Delta b). \quad (6)$$

Smooth L1 [19] is defined as follows:

$$\text{SmoothL1} = \begin{cases} 0.5x^2 & |x| < 1 \\ |x| - 0.5 & |x| > 1 \end{cases}. \quad (7)$$

The position regression loss function incorporates orientation regression loss. If only the angle difference  $\Delta \theta = \theta^{\text{gt}} - \theta^a$  is computed, situations can arise where the position prediction is correct but the orientation is opposite, leading to a situation where the angle loss is excessively high and affects the overall loss function value  $\mathfrak{L}_{\text{loc}}$ . Therefore, the sine function is applied to

the angle difference to address this issue. However, solely using the sine function may not distinguish and penalize two predicted angle values that are opposite in orientation. The algorithm proposed in this paper introduces an orientation regression classifier, defining two directions as positive and negative, constraining the angles within the  $(0, 2\pi)$  interval, mapping angles in the range  $[0, \pi)$  to 0 and angles in the range  $(\pi, 2\pi)$  to 1, representing them using one-hot encoding [20]. Subsequently, it employs the softmax and cross-entropy loss functions to compute another orientation regression loss value, denoted as  $\mathfrak{L}_{\text{dir}}$ .

In summary, the entire algorithm model's loss function is composed of as follows:

$$\mathfrak{L} = \frac{1}{N_{\text{pos}}} (\beta_{\text{loc}} \mathfrak{L}_{\text{loc}} + \beta_{\text{cls}} \mathfrak{L}_{\text{cls}} + \beta_{\text{dir}} \mathfrak{L}_{\text{dir}}), \quad (8)$$

where  $N_{\text{pos}}$  represents the number of positive samples, and in training, the values for  $\beta_{\text{loc}}$ ,  $\beta_{\text{cls}}$ , and  $\beta_{\text{dir}}$  are set to 2, 1, and 0.5, respectively.

The training process adjusts the batch size based on the GPU configuration of the training platform. A larger batch size means that a greater number of samples are fed into the network at once, allowing the determined directions to better represent the overall dataset. However, it also requires a larger GPU memory capacity. Indeed, with a larger batch size, while keeping the overall dataset size constant, there will be fewer iterations per epoch. Therefore, it's necessary to increase the number of epochs to achieve more iterations and potentially better results.

### 3. Model Evaluation and Deployment

**3.1. Model Optimization Evaluation Results.** The final detection results of the PyTorch Model are Shown in Figure 8.

From Figure 8, it can be observed that the model performs well in detecting vehicles of different classes, even when they are occluded or partially visible. According to the analysis of the test dataset, the model also exhibits a certain degree of dust-resistant detection capability, with high confidence in vehicle detection, which makes it suitable for vehicle detection in complex and dynamic environments such as open-pit mines. The effect of dust suppression detection is shown in Figure 9.

The loss function curve during model training is depicted in Figure 10.

To evaluate the performance of the PointPillars-HSL model with optimized feature preprocessing, a comparison was made against the PointPillars model without pretrained parameters on the same batch of mining area dataset, using identical batch size and learning rate strategy. The detection metrics on the open-pit mine dataset are shown in Table 1.

To verify the added pillar internal density features and  $z$ -direction point cloud distribution features, an ablation study was designed, as shown in Table 2. Feature Enhancement means adding density features and  $z$ -direction point cloud distribution features to the original 10D features. We can see that the model's mean average precision (mAP) for all classes improves about 8.13% for added 2D features.

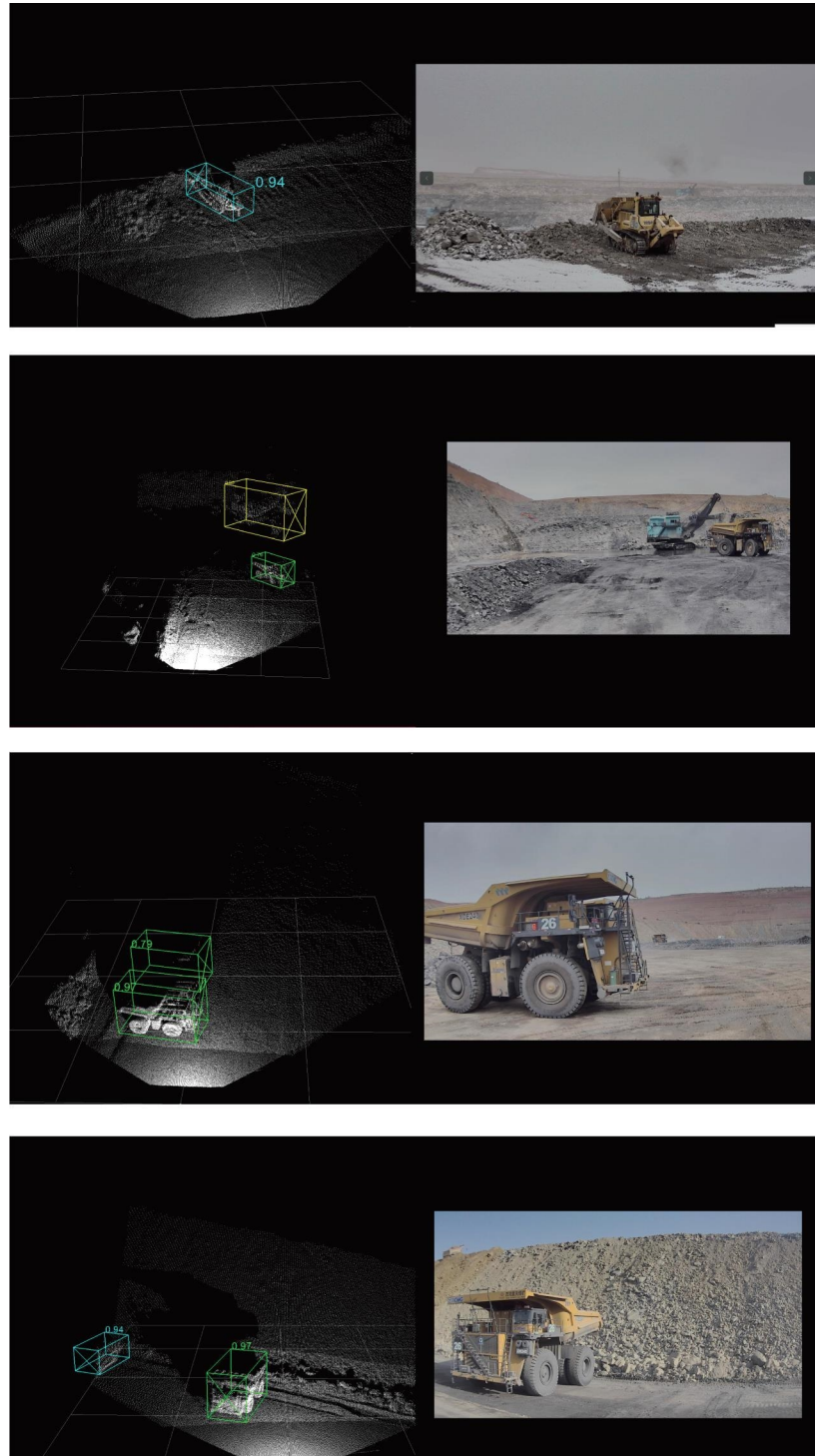


FIGURE 8: Mining truck detection performance.

To verify the impact of pillar grid division on model detection accuracy and speed, an ablation study, as shown in Table 2, was designed, and the detection results were measured on a device equipped with an Nvidia GeForce RTX 3050 Ti using the test dataset. The PointPillars model takes about 98.5 ms (10.15 Hz) to process each sample, and PointPillars-HSL uses a novel pillar grid division method to reduce the detection time by 11.3 ms without significant loss of accuracy.

**3.2. Model Deployment.** The model is deployed on an NVIDIA Tegra Xavier. After training, the PyTorch model file is converted to an ONNX file, and then inference acceleration is achieved using TensorRT to obtain the detection results.

The detection results of the PyTorch model, as shown in Figure 11, are based on a coordinate system defined in the OpenPCDet framework: with the positive  $x$ -axis representing



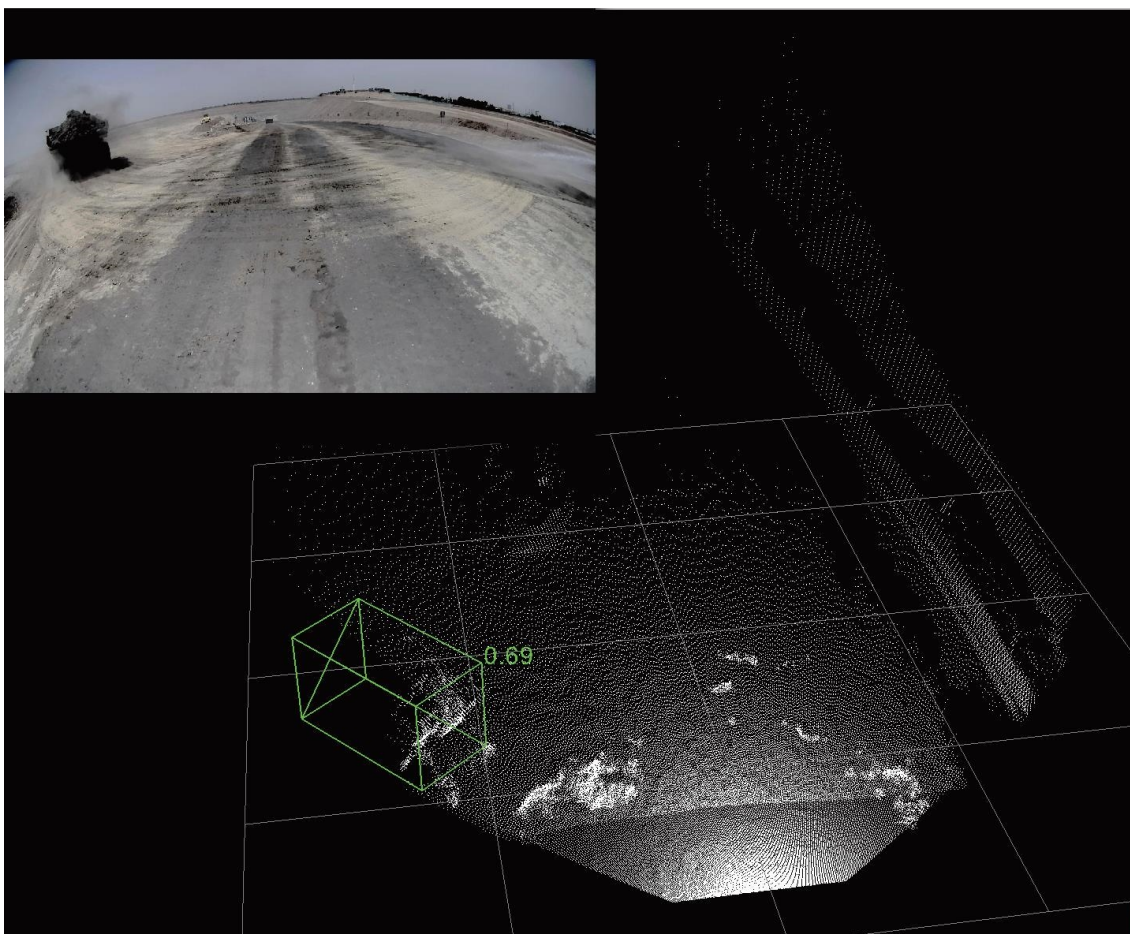


FIGURE 9: The effect of dust suppression detection.

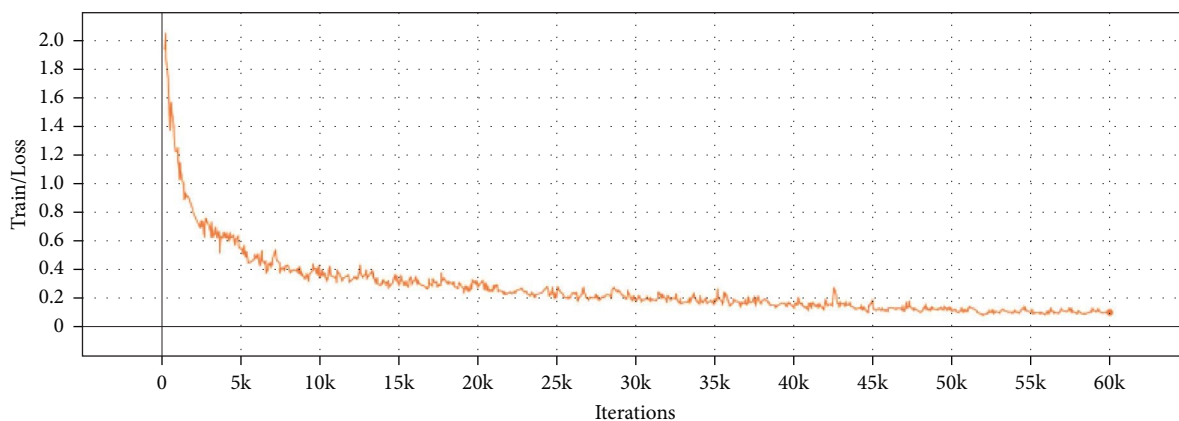


FIGURE 10: The loss function curve.

the forward direction, clockwise as positive, and angles ranging from  $(-\pi, \pi)$ .

In contrast, the reference coordinate system for the deployed inference results, as shown in Figure 12, is consistent with the training annotation data reference system. It uses the rightward direction as a reference, clockwise as positive, and also has an angle range of  $(-\pi, \pi)$ . The two coordinate systems differ by  $90^\circ$ .

The inference detection speed on the NVIDIA Tegra Xavier development board is approximately 65 ms/frame.

#### 4. Discussion

In summary, our research is the first to implement a sensing solution based on hybrid solid-state LiDAR in an open-pit mining scene. Due to the low probability of auxiliary vehicles

TABLE 1: The effectiveness of model feature optimization.

3D AP (%)	Mining truck	Bulldozer	Car	mAP
PointPillars	83.63	78.49	75.73	79.28
PointPillars-HSL	93.86	90.72	84.58	89.72

TABLE 2: Ablation study for PointPillars-HSL.

Model	Feature enhancement	Pillar size	Loss function optimization	mAP (%)	Speed (Hz)
PointPillars	X	0.162	X	79.28	10.15
	✓	0.162	X	87.41	—
PointPillars-HSL	X	0.162 + 0.322	X	79.05	11.47
	✓	0.162 + 0.322	X	87.21	—
	✓	0.162 + 0.322	✓	89.72	11.47

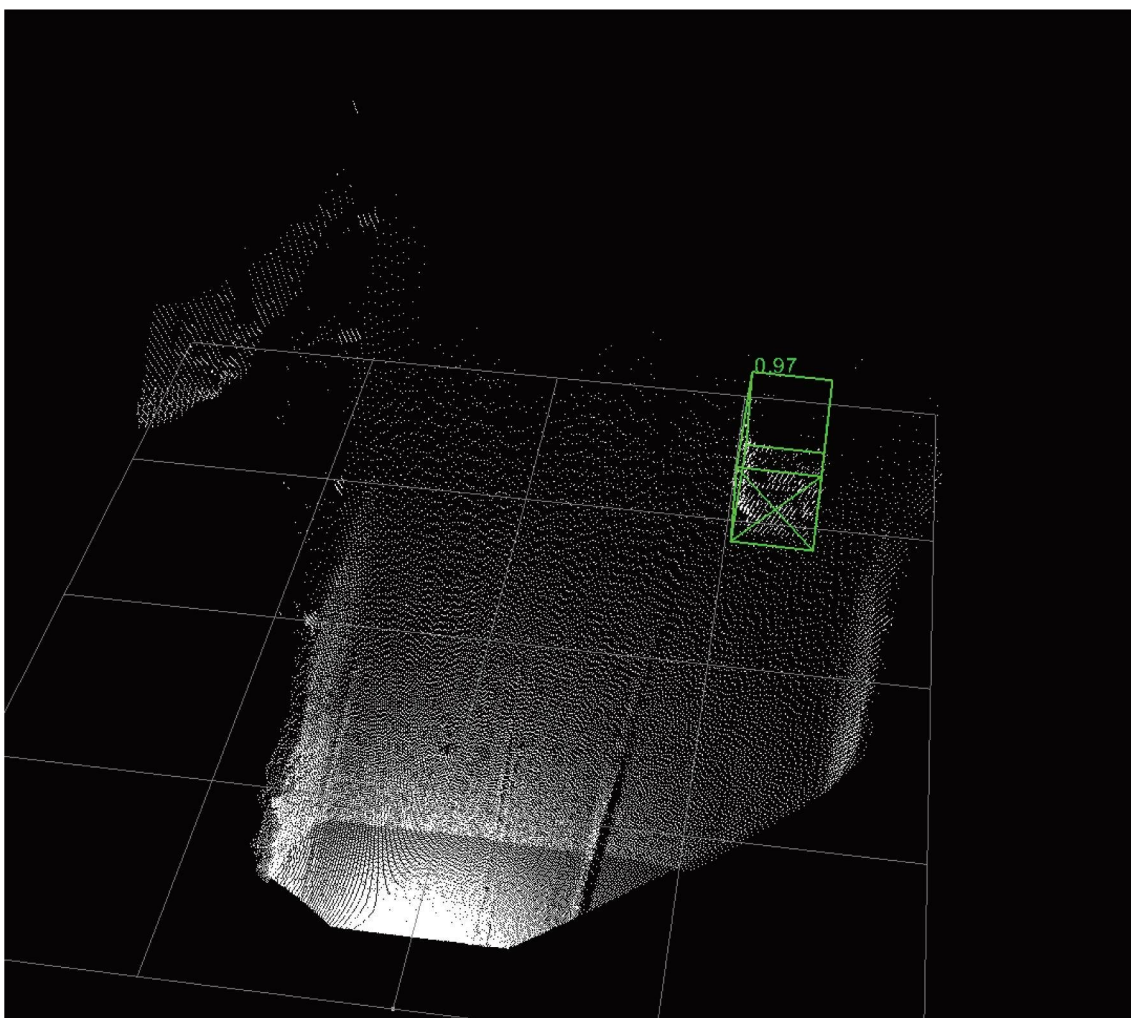


FIGURE 11: Detection results of PyTorch model.

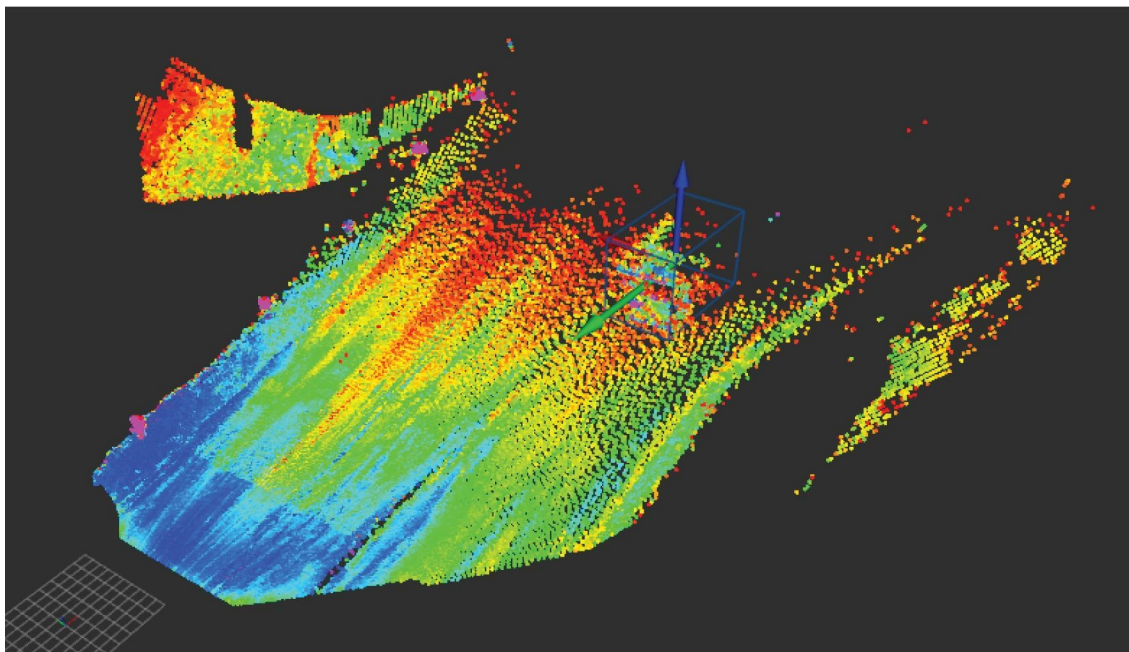


FIGURE 12: Detection results in Xavier.

such as sprinklers and command vehicles appearing in open-pit mines, sufficient data are lacking. In the future, with enough data, the detection solution can cover all types of vehicles in the mine. Combined with the traditional point cloud detection and segmentation algorithm, it can detect irregular objects such as stones and walls, taking the autonomous driving of mining trucks one step further.

## 5. Conclusions

This paper addresses the challenges faced in the perception of autonomous driving environments in mining, characterized by harsh operating conditions and complex, ever-changing road environments. We propose the PointPillars-HSL 3D object detection algorithm, which is suitable for mining environments and utilizes a hybrid solid-state LiDAR system. After analyzing the data from the hybrid solid-state LiDAR point cloud and by implementing downsampling and feature optimization based on Pillars, along with data preprocessing and transfer learning techniques, we have effectively addressed the issues of overly dense point clouds in open-pit mine and the significant slopes in unstructured road scenes. Through the optimization of the loss function, the stability of the model in predicting obstacle orientation and category has been enhanced. Furthermore, deploying the well-trained algorithm model on the NVIDIA Tegra Xavier has enabled real-time inference for 3D point cloud object detection.

## Data Availability

Data is available upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Authors' Contributions

Conceptualization was done by Cheng Li and Teng Long; software-related task was done by Cheng Li; validation was done by Gang Yao and Peijie Li; investigation was done by Gang Yao; resources were provided by Xiwen Yuan; data curation was done by Gang Yao and Peijie Li; writing—original draft preparation was done by Cheng Li; writing—review and editing was done by Teng Long; visualization was done by Cheng Li and Gang Yao; project administration was done by Xiwen Yuan. All authors have read and agreed to the published version of the manuscript.

## Acknowledgments

We sincerely appreciate CRRC Zhuzhou Institute Co., Ltd. for providing valuable experimental opportunities for this study. We also deeply value the significant contributions of our esteemed reviewers, whose prompt and insightful evaluations have enhanced the scientific value of the research. This research was funded by the National Key Research and Development Program of China, grant number 2022YFB4300405.

## References

- [1] S. Mittal and Vibhu, "A survey of accelerator architectures for 3D convolution neural networks," *Journal of Systems Architecture*, vol. 115, Article ID 102041, 2021.
- [2] A. Novak, A. N. Sedlackova, M. Vochozka, and G. H. Popescu, "Big data-driven governance of smart sustainable intelligent transportation systems: autonomous driving behaviors, predictive modeling techniques, and sensing and computing technologies," *Contemporary Readings in Law and Social Justice*, vol. 14, no. 2, pp. 100–117, 2022.
- [3] 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, Providence, RI, USA, 2012.
- [4] J. Behley, M. Garbade, A. Milioto et al., “Semantickitti: a dataset for semantic scene understanding of lidar sequences,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9297–9307, IEEE, 2019.
- [5] J. Xue, J. Fang, T. Li et al., “BLVD: building a large-scale 5D semantics benchmark for autonomous driving,” in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6685–6691, IEEE, 2019.
- [6] 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, IEEE, 2017.
- [7] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “Pointnet++: deep hierarchical feature learning on point sets in a metric space,” vol. 30, *Advances in neural information processing systems*, 30, 2017.
- [8] Y. Zhou and O. Tuzel, “VoxelNet: end-to-end learning for point cloud based 3D object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4490–4499, IEEE, 2018.
- [9] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, “Joint 3D proposal generation and object detection from view aggregation,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1–8, IEEE, Madrid, Spain, 2018.
- [10] 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, IEEE, 2019.
- [11] J. Stanisiz, K. Lis, and M. Gorgon, “Implementation of the pointpillars network for 3D object detection in reprogrammable heterogeneous devices using FINN,” *Journal of Signal Processing Systems*, vol. 94, no. 7, pp. 659–674, 2022.
- [12] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, “Frustum pointnets for 3D object detection from RGB-D data,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 918–927, IEEE, Salt Lake City, UT, USA, 2018.
- [13] B. Yang, M. Liang, and R. Urtasun, “HDNET: exploiting HD maps for 3D object detection,” in *Conference on Robot Learning*, pp. 146–155, PMLR, 2018.
- [14] Y. Yan, Y. Mao, and B. Li, “SECOND: sparsely embedded convolutional detection,” *Sensors*, vol. 18, no. 10, Article ID 3337, 2018.
- [15] F. Zhuang, Z. Qi, K. Duan et al., “A comprehensive survey on transfer learning,” *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.
- [16] Z. Tian, C. Shen, H. Chen, and T. He, “FCOS: a simple and strong anchor-free object detector,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 4, pp. 1922–1933, 2020.
- [17] W. Liu, D. Anguelov, D. Erhan et al., “SSD: single shot multiBox detector,” in *Computer Vision—ECCV 2016*, vol. 9905 of *Lecture Notes in Computer Science*, pp. 21–37, Springer International Publishing, Cham, 2016.
- [18] P. Yun, L. Tai, Y. Wang, C. Liu, and M. Liu, “Focal loss in 3D object detection,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1263–1270, 2019.
- [19] I. Onaran, N. F. Ince, and A. E. Cetin, “Sparse spatial filter via a novel objective function minimization with smooth 1 regularization,” *Biomedical Signal Processing and Control*, vol. 8, no. 3, pp. 282–288, 2013.
- [20] L. Jie, C. Jiahao, Z. Xueqin, Z. Yue, and L. Jiajun, “One-hot encoding and convolutional neural network based anomaly detection,” *Journal of Tsinghua University (Science and Technology)*, vol. 59, no. 7, pp. 523–529, 2019.