Scilight

Sensors and AI

https://www.sciltp.com/journals/sai



Article

Boundary-Aware Semantic Segmentation of Pig Point Clouds in Commercial Housing Environments

Zhankang Xu [1,2,†], Fei Shi [1,3,†], Xiangyu Qi [1], Zhaoyang Wang [1], Mengxin Guo [1], Yikai Fan [1], Simon X. Yang [4], Qifeng Li [1,*] and Weihong Ma [1,4,*]

[1] Information Technology Research Center, Beijing Academy of Agriculture and Forestry Sciences, Beijing 100097, China
[2] Bureau of Agriculture and Rural Affairs of Shangrao City, Shangrao 334000, China
[3] School of Science, China University of Geosciences (Beijing), Beijing 100083, China
[4] Advanced Robotics and Intelligent Systems Laboratory, School of Engineering, University of Guelph, Guelph, ON N1G 2W1, Canada
* Correspondence: nercita1017@163.com (Q.L.); mawh@nercita.org.cn (W.M.)
† These authors contributed equally to this work.

How To Cite: Xu, Z.; Shi, F.; Qi, X.; et al. Boundary-Aware Semantic Segmentation of Pig Point Clouds in Commercial Housing Environments. *Sensors and AI* **2026**, *2*(1), 50–71. https://doi.org/10.53941/sai.2026.100004

Received: 23 June 2026
Revised: 15 October 2026
Accepted: 4 November 2026
Published: 18 November 2026

Abstract: In commercial pig-housing environments, contact between pig-body point clouds and background structures can produce blurred boundaries, local adhesion, and background misclassification, which may propagate to downstream three-dimensional phenotyping. This study proposes a boundary-aware pig point-cloud segmentation method based on an Octree Transformer backbone. The framework combines multi-scale feature fusion, soft-distance boundary pseudo-labels, and bidirectional semantic-boundary interaction. On the prespecified independent test split, the method achieved an accuracy (*Acc*) of 98.65%, a mean intersection over union (*mIoU*) of 97.36%, and a boundary intersection over union (*B-IoU*) of 88.77%. These are single-run point estimates; inferential significance and variability across random initializations were not assessed. The evaluation was intended to examine the complete boundary-aware workflow under a fixed independent animal-level protocol rather than to establish statistical significance from small numerical margins. The observed advantage was largest for *B-IoU*, consistent with the method's explicit boundary objective, and should therefore be interpreted together with *Acc*, *mIoU*, and qualitative results. The findings support the method under the two sampled farms and fixed acquisition configurations, while cross-farm and cross-device generalization remain to be established.

Keywords: pig point cloud segmentation; boundary-aware learning; Octree Transformer; soft-distance boundary pseudo-label; precision livestock farming

1. Introduction

With the development of three-dimensional (3D) vision, deep learning, and intelligent perception technologies, point clouds—capable of directly representing the spatial geometric structure of objects—have been widely applied in 3D reconstruction, object recognition, scene understanding, and smart agriculture. Compared with two-dimensional (2D) images, point clouds not only record the 3D coordinates of target surfaces but also capture object contours, local surface details, and spatial topological relationships. Consequently, they are particularly suitable for acquiring animal phenotypic information and measuring body size parameters [1–3]. In pig farming scenarios, acquiring pig body point clouds using RGB-D cameras or multi-view depth cameras provides a critical data foundation for monitoring growth, assessing health, analyzing production performance, and performing non-contact body size measurements [4,5].

Copyright: © 2026 by the authors. This is an open access article under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).
Publisher's Note: Scilight stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

However, original point clouds captured in real pigsty environments typically include not only pig bodies but also background structures such as the floor, fences, railings, and data acquisition devices. As pigs are non-rigid living targets, postural changes—including head turning, walking, pausing, and approaching railings—frequently occur during data collection, often leading to local adhesion between pig body point clouds and background point clouds. Additionally, occlusion, sensor noise, uneven point cloud density, and background interference further compromise the geometric consistency of pig body boundaries, resulting in blurred class transitions, complex local structures, and unstable feature distributions at the boundaries. Residual background points or missing pig-body edges may influence subsequent processes such as point-cloud completion, keypoint detection, and body-size measurement, potentially reducing the accuracy of three-dimensional phenotypic analysis. Consequently, fine segmentation of pig point clouds under complex backgrounds is an important pre-processing step for automated acquisition of three-dimensional phenotypes.

Early methods for animal point cloud segmentation primarily relied on strategies such as background modeling, Euclidean clustering, RANSAC plane removal, and manually defined geometric rules. These approaches can achieve satisfactory results when the target and background are clearly separated and the acquisition environment is relatively controlled [4,6,7]. However, such methods typically depend on fixed thresholds or hand-crafted features, limiting their adaptability to variations in pig postures, railing occlusions, point cloud density fluctuations, and boundary adhesion. In recent years, deep learning has facilitated the transition of point cloud segmentation from rule-driven approaches to data-driven feature learning. PointNet and PointNet++ laid the foundation for directly processing unordered point sets [8,9], while methods such as DGCNN, KPConv, Point Transformer, PointNeXt, and OctFormer further enhanced the modeling of local geometric relationships, long-range dependencies, and multi-scale structures in point clouds [10–14]. In the context of livestock and poultry point clouds, Lin et al. [15] employed improved DBSCAN and PointNet++ to extract pig point clouds, Wang et al. [16] combined 2D segmentation results with depth information to acquire pig body point clouds, and Chang et al. [17] enhanced segmentation of pig body and railing point clouds in complex pig farms by improving PointNet++ with SoftPool. Beyond livestock-specific applications, boundary-aware point-cloud semantic segmentation has been investigated through boundary feature analysis, contrastive boundary learning, and category-guided feature aggregation [18–20]. These studies demonstrate that explicit modeling of boundary structure can improve feature discrimination near semantic transitions in general three-dimensional scenes. However, pig point clouds acquired in commercial housing environments present a specific form of boundary ambiguity caused by physical contact and spatial adhesion between non-rigid animal surfaces and background structures such as floors, fences, and railings. Building on the broader development of boundary-aware segmentation, the present study investigates the joint use of continuous distance-decayed boundary supervision and bidirectional semantic-boundary interaction within an OctFormer framework for these livestock-specific contact regions.

Existing pig point-cloud studies have reported background inclusion, pig-body edge omission, and local adhesion near physical pig-background contact regions [17,21]. More broadly, boundary feature analysis has demonstrated the value of explicit boundary modeling for three-dimensional semantic segmentation [18]. Motivated by these findings, the present study jointly employs continuous boundary supervision and semantic-boundary interaction to improve segmentation in ambiguous contact regions.

To address these issues, this paper proposes a pig point cloud segmentation method based on boundary feature analysis. The proposed method adopts Octree Transformer as the backbone network, organizes the original point cloud using an octree-based hierarchical structure, and incorporates multi-scale feature extraction and cross-layer fusion mechanisms to model both local geometric details and global semantic context in pig point clouds. Furthermore, soft-distance boundary pseudo-labels are generated to provide continuous and smooth boundary supervision signals for contact regions between the pig body and the background. A bidirectional cross-boundary semantic module is further designed to enable explicit interaction between boundary features and semantic features, thereby enhancing the model's discriminative capability for complex contact boundaries. The main contributions of this paper are summarized as follows:

(1) To address the problem of boundary adhesion between pig body and background point clouds in complex pigsty environments, a pig point cloud segmentation framework based on Octree Transformer was developed to jointly capture local geometric details and global semantic context.
(2) A soft-distance boundary pseudo-label is proposed to convert discrete boundary supervision into continuous boundary constraints, aiming to mitigate training instability caused by fuzzy category transitions near boundaries and variations in point cloud density.
(3) A bidirectional cross-boundary semantic module is designed to facilitate interaction and enhancement between semantic features and boundary features, thereby improving the model's fine-grained segmentation capability for contact regions between the pig body and background structures such as the floor, railings, and fences.

2. Materials and Methods

2.1. Data Acquisition

The point-cloud data were collected from 440 Duroc × Landrace × Large White crossbred pigs at Tieqilishi Pig Farm in Mianyang, Sichuan Province, and a commercial farm in Longyao County, Xingtai, Hebei Province. Body weights ranged from 25 to 150 kg. The cohort size was determined pragmatically by the animals available during the scheduled farm campaigns, subject to non-invasive collection, successful radio-frequency identification (RFID) association, and post-acquisition quality screening. No prospective power calculation was performed because the primary outcomes were point-wise segmentation metrics rather than an animal-level treatment effect. Data collection occurred before daily feeding, between 06:30 and 08:30.

The acquisition zone was located in the main passage of a partitioned barn and included directional guide fences, infrared photoelectric triggers, and a RFID reader. Interruption of the infrared beam triggered synchronized depth acquisition, while the RFID system associated each acquisition with an animal ID [22]. Only XYZ coordinates were retained; RGB channels were omitted to reduce storage and sensitivity to illumination. Point-cloud phenotyping in typical swine pens has also been used for downstream body-mass prediction, underscoring the importance of reliable geometric preprocessing and foreground extraction [23].

The dataset was collected from two commercial pig-farm sites with relatively stable acquisition conditions. Although both sites used comparable passage-based acquisition layouts, differences in housing materials, floor appearance, railing morphology, lighting, and local background structure may still exist between farms. These site-specific factors were retained in the collected data and were not independently controlled through a leave-one-farm-out validation protocol. The overall data acquisition system and on-site experimental setup are shown in Figure 1.

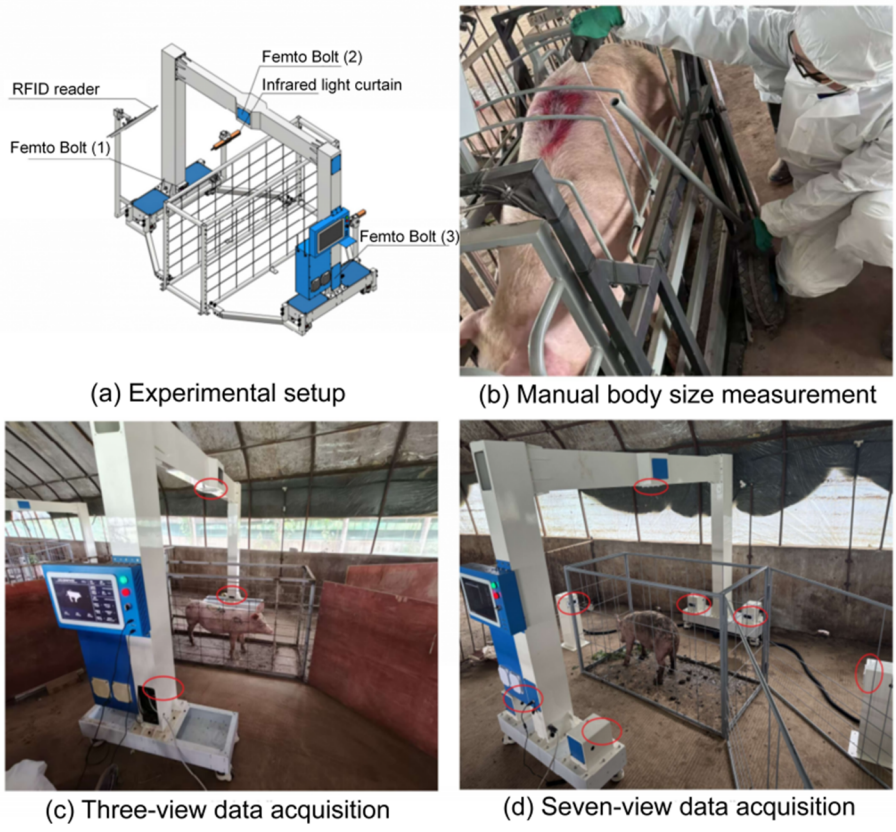

Figure 1. Data acquisition system and field setup for pig point cloud collection. The system integrates multi-view depth cameras, infrared triggering, directional guide fences, and RFID-based individual identification for synchronized point cloud acquisition in a commercial pig housing environment.

To balance practical acquisition efficiency and the integrity of point cloud structures, this study developed two types of synchronous point cloud acquisition systems with three-view and seven-view configurations. The three-view system consists of three Orbbec Femto Bolt depth cameras (Orbbec Inc., Shenzhen, China), which are arranged above and on the left and right sides of the pig body, respectively, forming a spatial observation structure from the upper and bilateral directions. The cameras operate in WFOV 2 × 2 binned modes, with a field of view

of 120° × 120°, an effective measurement range of 0.25–2.88 m, and an acquisition frame rate of 30 fps. The distance from the lens of the top camera to the ground is approximately 1.6 m, the distance between the lenses of the two side cameras is approximately 2.0 m, and the width of the acquisition channel is approximately 0.5 m, allowing pigs to pass through in a single file. The three-view system can acquire point clouds of the dorsal and lateral regions of the pig body; however, local missing areas may still occur in the abdomen, the inner sides of the limbs, and regions occluded by railings. An example of a three-view pig point cloud is shown in Figure 2.

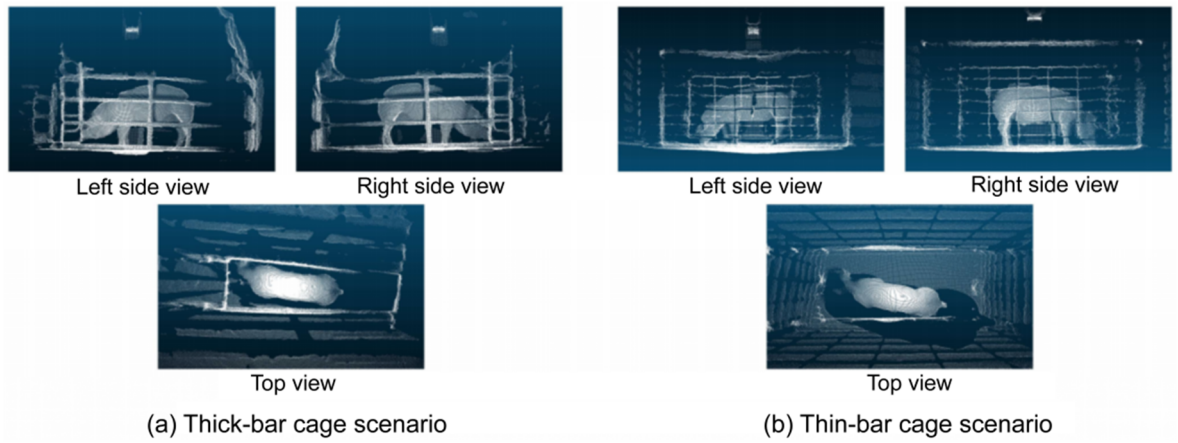

Figure 2. Representative three-view pig point clouds under two cage scenarios: (**a**) thick-bar cage scenario. (**b**) thin-bar cage scenario. The three-view configuration captures dorsal and bilateral surface information of the pig body, while local missing regions may remain in occluded areas such as the abdomen, inner limbs, and railing-contact regions.

To further mitigate the impact of single-view occlusion on the integrity of pig point clouds, this study developed a seven-view synchronous point cloud acquisition system. The system comprises seven depth cameras arranged according to an optimized spatial layout, which synchronously capture pig point clouds from the top, left front, right front, left rear, right rear, front, and rear directions, forming a multi-directional redundant observation structure. The camera models, acquisition parameters, and triggering methods of the seven-view system are consistent with those of the three-view system, and hardware synchronization ensures that multi-camera data are acquired within the same time window, thereby minimizing the effects of slight pig movements on point cloud fusion. Compared with three-view data, seven-view point clouds exhibit higher coverage integrity and spatial continuity in typically occluded regions such as the abdomen, inner sides of the limbs, head, and buttocks, enabling a more comprehensive representation of the overall geometric shape of the pig body. During data acquisition, each pig passes through the passage 2–3 times to increase sample size and pose diversity. An example of a seven-view pig point cloud is shown in Figure 3.

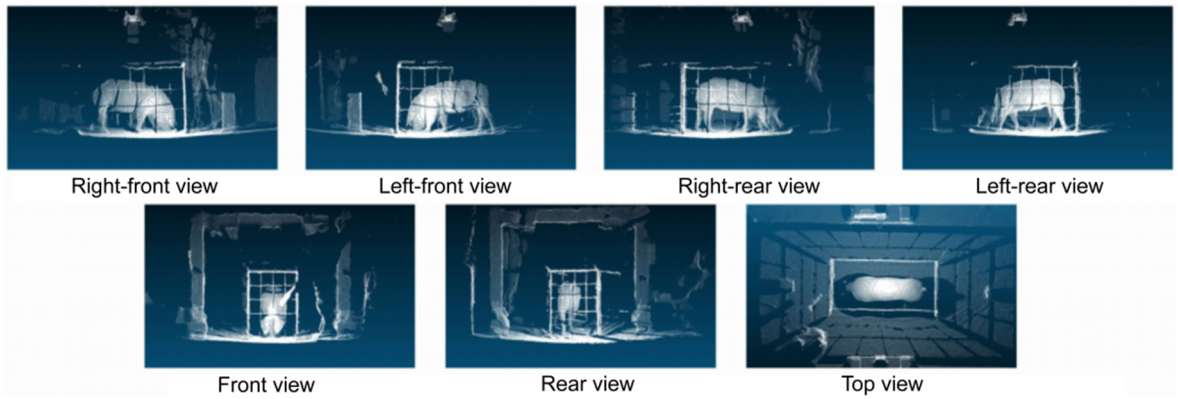

Figure 3. Representative seven-view pig point cloud. The seven-view configuration provides more complete spatial coverage of the pig body by capturing point clouds from the top, front, rear, and bilateral oblique directions.

2.2. Point Cloud Preprocessing and Annotation

To fuse multi-view point clouds within a unified coordinate system, this study adopted Zhang's checkerboard calibration method [24] to geometrically calibrate multiple synchronized cameras. During calibration, a 60 cm ×

80 cm black-and-white checkerboard was used, and multiple sets of images with different poses were collected under unobstructed conditions. First, the focal length, principal point coordinates, and distortion parameters of each camera were estimated. Then, the extrinsic parameter relationships between cameras were obtained through pairwise chain calibration. The coordinate system of the top camera was used as the global reference coordinate system. Point clouds captured from different viewpoints were transformed into the unified world coordinate system, with the spatial unit standardized to millimeters (mm) in the saved point-cloud files. In this coordinate system, the Z-axis is perpendicular to the ground, the X-axis follows the direction of the acquisition channel, and the Y-axis is parallel to the ground and perpendicular to the channel direction. Through the above calibration and coordinate transformation procedures, multi-view fused point clouds with consistent spatial positions were obtained, providing standardized inputs for subsequent region of interest (ROI) selection, segmentation annotation, and model training. The original calibration records did not preserve a verified numerical reprojection error, per-camera calibration residual, or 3D registration RMSE. Accordingly, no numerical calibration-accuracy value is reported in the present study. The checkerboard calibration and the resulting intrinsic and extrinsic parameters were used to establish the common coordinate system for multi-view point-cloud fusion. A quantitative calibration-error assessment should be included in future data-collection campaigns.

The available acquisition records indicate that the depth cameras were hardware-triggered within the same acquisition window after a pig interrupted the infrared photoelectric barrier. This acquisition procedure was intended to reduce temporal misalignment caused by pig movement during multi-view point-cloud fusion. However, the original records did not preserve a verified numerical synchronization error or maximum inter-camera time offset. Therefore, the present study does not report a numerical synchronization-accuracy value.

After multi-view point-cloud fusion, spatial range cropping and quality screening were performed on all samples. First, a fixed ROI was defined according to the physical dimensions of the acquisition channel. Redundant background points clearly outside the pigs' activity range were removed, including points below the ground plane, points above the usable passage height, and isolated points outside the lateral and longitudinal limits of the acquisition channel. The same ROI rule was applied to all samples before dataset partitioning. The ROI was defined according to the physical acquisition volume rather than the semantic labels and was not used to isolate the pig body. Floor surfaces, fences, railings, and other structures located within the animal-accessible passage were retained, including background points adjacent to or in physical contact with the pig body. Therefore, the cropped point clouds continued to contain the principal foreground-background ambiguities encountered during practical acquisition. A separate numerical record of the retained background-point proportion was not available, and no retrospective percentage is reported. The original processing records did not preserve the numerical ROI thresholds in the global coordinate system. Therefore, the ROI is reported as a deterministic physical selection rule rather than through reconstructed coordinate values. Subsequently, manual quality inspection was conducted on the point-cloud samples, and distorted data caused by severely bent postures, jumping, motion blur, incomplete acquisition, or sensor abnormalities were excluded. After screening, a total of 3392 point-cloud samples were obtained from the three-view and seven-view data, among which 1042 were complete seven-view point clouds. The remaining 2350 samples were acquired using the three-view configuration. All point clouds retained the corresponding pig IDs to ensure that data from the same pig under different views and at different acquisition times could be tracked and managed.

For reproducibility, the model input consisted of XYZ coordinates only; RGB and intensity features were not used. No explicit mean-variance standardization or intensity normalization was applied. Each point cloud was recentered using a coordinate-shift operation. During training, random point dropout was applied with a dropout ratio of 0.2 and an application probability of 0.2. Additional augmentation included random rotations around the x-, y-, and z-axes, random scaling in the range [0.9, 1.1], random flipping, and coordinate jittering with a jitter standard deviation of 5 coordinate units and a clipping value of 20 coordinate units. The point clouds were voxelized using a grid size of 2.0, randomly cropped using sphere cropping with a sample rate of 0.8, and limited to a maximum of 120,000 points. These model-side augmentation operations were applied only to the training set; validation and test samples were processed deterministically.

The point-cloud segmentation dataset was manually annotated using CloudCompare software (Version 2.13.2). Since the original point clouds contained both pig bodies and environmental structures, including the ground, fences, railings, and data-acquisition devices, the segmentation task was defined as a binary classification problem: pig-body points were assigned the label "1", and environmental-background points were assigned the label "0". The annotation protocol specified the treatment of the main pig-body surface, background structures, and contact regions where the pig body was adjacent to the floor, fences, or railings. Particular attention was given to preserving the visible pig-body contour while avoiding the inclusion of clearly identifiable background points.

During annotation, each fused point cloud was inspected from multiple viewing angles in CloudCompare. Points that were geometrically continuous with the visible surfaces of the pig's head, trunk, or limbs were assigned to the pig-body class, whereas points continuous with the floor, fences, railings, or acquisition equipment were assigned to the background class. At physical contact regions, labels were determined by tracing local surface continuity from adjacent unambiguous regions, without interpolating unobserved pig-body points. The same decision rules were applied consistently to all samples.

Contact boundaries may remain intrinsically ambiguous because of occlusion, point-density variation, sensor noise, and physical contact between the pig body and background structures. All samples were therefore annotated by the same trained annotator using the same annotation protocol to improve procedural consistency and reduce variability caused by changes between annotators. This single-annotator procedure improves consistency within the dataset, but it does not provide a quantitative estimate of inter-annotator agreement or annotation repeatability. Because the resulting labels were used both to generate SDBPL supervision and to define the reference boundaries for *B-IoU*, possible annotation uncertainty was considered when interpreting boundary-specific results.

For dataset partitioning, the training, validation, and test sets were divided using individual pigs as the basic unit rather than by randomly splitting individual point-cloud samples. This strategy was adopted to avoid data leakage caused by samples from the same pig, captured from different viewpoints or at different acquisition times, appearing in different subsets. The animal-level allocation was 352, 44, and 44 pigs for the training, validation, and test sets, respectively, corresponding to an 8:1:1 split by pig count. The resulting point-cloud sample counts were 2096, 449, and 449 for the training, validation, and test sets, respectively. The available split records document the animal-level allocation but do not preserve a separate farm-by-split allocation table. Therefore, explicit farm-stratified sampling cannot be verified from the current records, and the reported test results should be interpreted as performance on the prespecified animal-level test split rather than as a leave-one-farm-out estimate of cross-farm generalization. The available body-weight information was considered when reviewing the representativeness of the subsets; however, no formal numerical balance analysis across body-weight ranges was retained. The detailed partitioning results are presented in Table 1.

Table 1. Dataset split for pig point cloud segmentation.

Dataset	Number of Pigs	Number of Point Cloud Samples	Number of Complete Seven-View Point Cloud Samples
Training set	352	2096	380
Validation set	44	449	81
Test set	44	449	81
Total	440	2994	542

Note: The dataset was split at the individual-pig level. "Number of complete seven-view point-cloud samples" refers to samples acquired using the seven-view system and retained after quality screening in each subset. The table reports the sample counts used in the documented training, validation, and test runs.

2.3. Boundary-Aware Pig Point Cloud Segmentation

2.3.1. Overview of the Proposed Method

In real pig-housing environments, pig point clouds may be adjacent to or partially merged with surrounding background structures. These contact regions often exhibit uneven point density, complex local geometry, and gradual semantic transitions. Conventional semantic segmentation networks may therefore produce boundary adhesion, local omissions, or background misclassification. To address these issues, this study proposes a boundary-aware pig point-cloud segmentation method.

The proposed method adopts Octree Transformer [14] as the backbone network. First, the input point cloud is organized into an octree-based hierarchical structure, and multi-scale point cloud features are extracted through octree convolution and self-attention encoding modules. Subsequently, coarse-scale global semantic information and fine-scale local geometric information are integrated using a cross-layer feature fusion mechanism. On this basis, soft-distance boundary pseudo-labels are generated to provide continuous boundary supervision for ambiguous contact regions. Finally, a bidirectional cross-boundary semantic module is employed to enable explicit interaction between semantic features and boundary features, and a joint loss function is adopted to simultaneously optimize semantic segmentation and boundary prediction. The overall network architecture is shown in Figure 4.

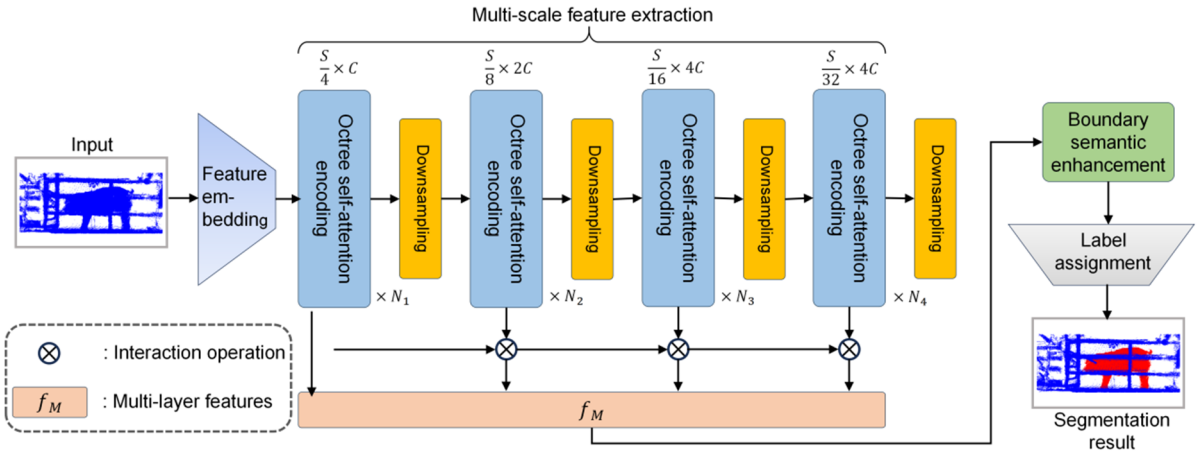

Figure 4. Overall architecture of the proposed boundary-aware pig point cloud segmentation network. The network uses Octree Transformer as the backbone and integrates multi-scale feature fusion, soft-distance boundary supervision, and bidirectional semantic-boundary feature interaction for pig point cloud segmentation.

2.3.2. Multi-Scale Point Cloud Feature Extraction

Point cloud data are characterized by unorderedness, sparsity, and irregular spatial distribution. Directly applying global self-attention to all points would incur substantial computational overhead. To improve the processing efficiency of large-scale pig point clouds, this study adopts the octree encoding strategy of OctFormer. Specifically, the original point cloud is first organized into an octree-based hierarchical structure, allowing adjacent non-empty nodes in three-dimensional space to participate in feature computation in a locally continuous manner.

In the feature embedding stage, the network employs five octree convolution modules to project the input point cloud features into a high-dimensional feature space. Each module consists of an octree convolution layer [25], batch normalization, and a ReLU activation function. The convolution kernel sizes are sequentially set to {3, 2, 3, 2, 3}, with corresponding strides of {1, 2, 1, 2, 1}. Among these modules, convolution layers with a stride of 2 are used for spatial downsampling. After five octree convolution layers, the point cloud features undergo an overall four-fold spatial downsampling, thereby reducing computational cost while enhancing local geometric representation capability.

In the octree self-attention encoding stage, the network is organized into multiple stages, each containing several octree self-attention encoding modules and downsampling modules. Each encoding module primarily comprises octree attention, a multi-layer perceptron (MLP), and layer normalization. To facilitate windowed attention computation, the Morton encoding strategy [26] is adopted to map adjacent non-empty octree nodes in three-dimensional space into a one-dimensional, locally continuous sequence. The sequence is then partitioned into fixed-length windows, within which self-attention is computed. To enhance cross-window information interaction, the network alternates dilation factors of D = 1 and D = 4 for window partitioning in consecutive self-attention layers. When D = 1, the model focuses primarily on detailed structures within local neighborhoods; when D > 1, the dilated windows span previously adjacent node blocks, thereby expanding the context receptive field. By alternately computing local and dilated windows, the model balances local boundary detail with large-scale semantic context while maintaining approximately linear computational complexity. The octree self-attention encoding module is illustrated in Figure 5.

Considering that different octree levels correspond to different semantic scales, this study further implements multi-scale feature fusion. In general, higher-level nodes capture stronger global structural information, while lower-level nodes retain richer local boundary details. Inspired by the multi-resolution feature interaction approach of HRNet [27], cross-layer fusion is performed on features from the last four octree levels. Following the settings of OctFormer, the maximum octree depth is 11, and therefore, fusion is applied from the 8th to the 11th levels. The computation of the fused feature f_M is defined as follows:

$$f_M = up(f_8) + \sum_{i=8}^{11} up\left(C^{3\times3}\left(up(f_i) \otimes C^{1\times1}(f_{i+1})\right)\right) \tag{1}$$

In the formula, f_i represents the features of the *i-th* octree layer; $\otimes$ denotes the feature interaction operation; $C^{1\times1}$ and $C^{3\times3}$ represent 1 × 1 and 3 × 3 convolution layers, respectively; and $up(\cdot)$ denotes the upsampling operation. Through cross-layer fusion, a point cloud representation capturing both global semantic information and fine-grained geometric structures can be obtained.

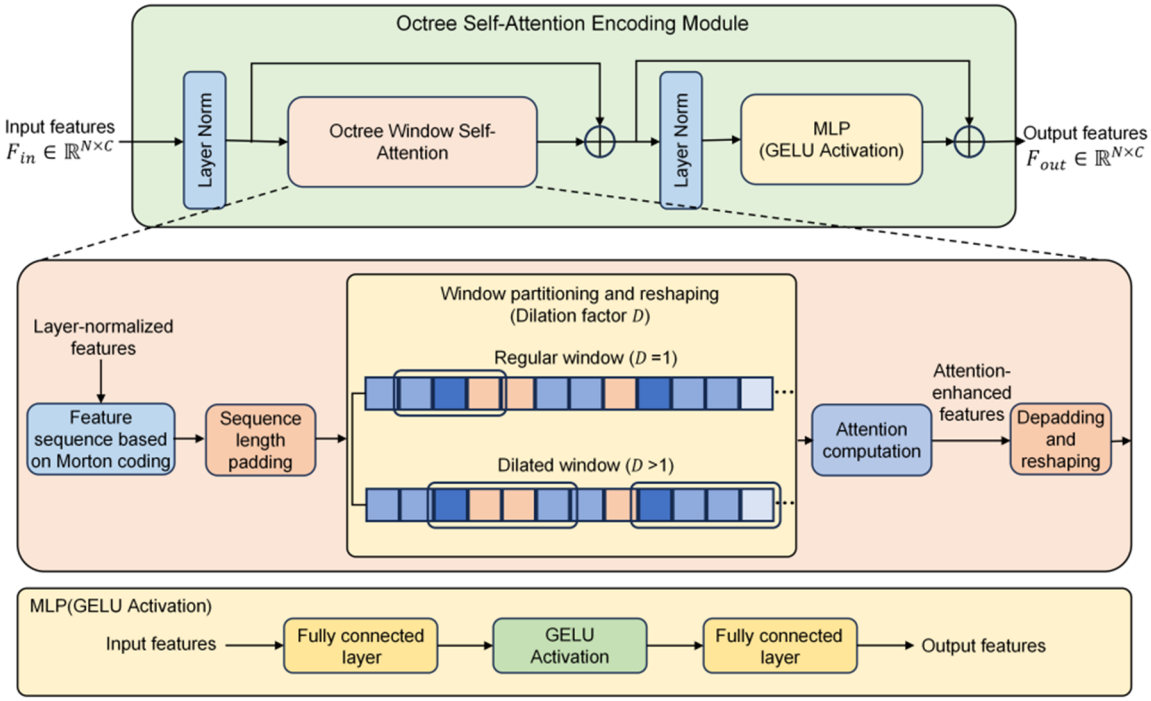

Figure 5. Architecture of the octree self-attention encoding module. Non-empty octree nodes are encoded using Morton ordering and processed through local and dilated window self-attention to capture both local geometric details and broader contextual information.

2.3.3. Soft-Distance Boundary Pseudo-Labeling

In pig point cloud segmentation, class boundaries between pig bodies and background structures, such as the floor, fences, and railings, are often not strictly discrete but instead exhibit spatial transition regions. Traditional hard-boundary pseudo-labels typically classify points simply as boundary or non-boundary points according to whether heterogeneous points are present in their local neighborhoods. Although this approach is straightforward to implement, it has two main limitations. First, boundary thickness is highly sensitive to point cloud sampling density. Second, binary labels can only distinguish boundary points from non-boundary points and cannot represent the continuous distance relationship between points and the true boundary, which hinders the model from learning the gradual structural variations near boundaries.

To address the aforementioned issues, this study introduces the Soft Distance Boundary Pseudo-Label (SDBPL). This approach generates a continuous boundary probability based on the distance from a point to its nearest out-of-class point, providing a smoother supervision signal for boundary regions. For any point i, let its spatial coordinates be C_i and its semantic label be S_i. Within the neighborhood $N_i(r)$ of radius r, the distance d_i from this point to its nearest out-of-class point is computed as follows:

$$d_i = \min_{j \in N_i(r), S_j \neq S_i} \left\| C_i - C_j \right\|_2 \tag{2}$$

In the formula, $\|\cdot\|_2$ denotes the Euclidean distance. If no out-of-class points exist within the neighborhood, let $d_i = +\infty$. Based on this distance, the soft-boundary pseudo-label $B_i^g \in [0, 1]$ is defined as follows:

$$B_i^g = \begin{cases} \exp\left(-\dfrac{d_i^2}{2\sigma_b^2}\right), & d_i \le r \\ 0, & else \end{cases} \tag{3}$$

In the formula, σ_b controls the width of the boundary transition zone. In this study, the boundary transition width was set to $\sigma_b = 25$ coordinate units, corresponding to 25 mm under the millimetre-scale coordinate convention of the dataset. The neighborhood radius used to search for the nearest out-of-class point was set to $r = 100$ coordinate units, corresponding to 100 mm (0.1 m). These values were fixed in the implementation configuration before the final test evaluation and were kept unchanged across all reported experiments. The radius was selected to define a local neighborhood around pig-background contact regions at the physical scale represented in the acquired point clouds, while σ was set to one quarter of the neighborhood radius to provide a gradual decay of boundary confidence within this local region. These settings were used as fixed scale parameters rather than values claimed to be globally optimal.

No exhaustive grid search or formal validation-set sensitivity analysis was completed for these parameters in the present study. Therefore, these values are reported as fixed experimental settings rather than as globally optimal values. As di approaches 0, point *i* becomes closer to the true boundary and its boundary probability approaches 1. As d_i increases, the boundary probability gradually decays toward 0. Compared with binary hard-boundary labels, the soft-distance boundary pseudo-labels form a continuous and differentiable constraint near boundaries. This formulation preserves boundary-position information and provides effective supervision for transition regions, thereby enhancing the model's ability to represent complex boundary contours involving the abdomen, limbs, and railings. A schematic diagram of the soft-distance boundary pseudo-label is illustrated in Figure 6.

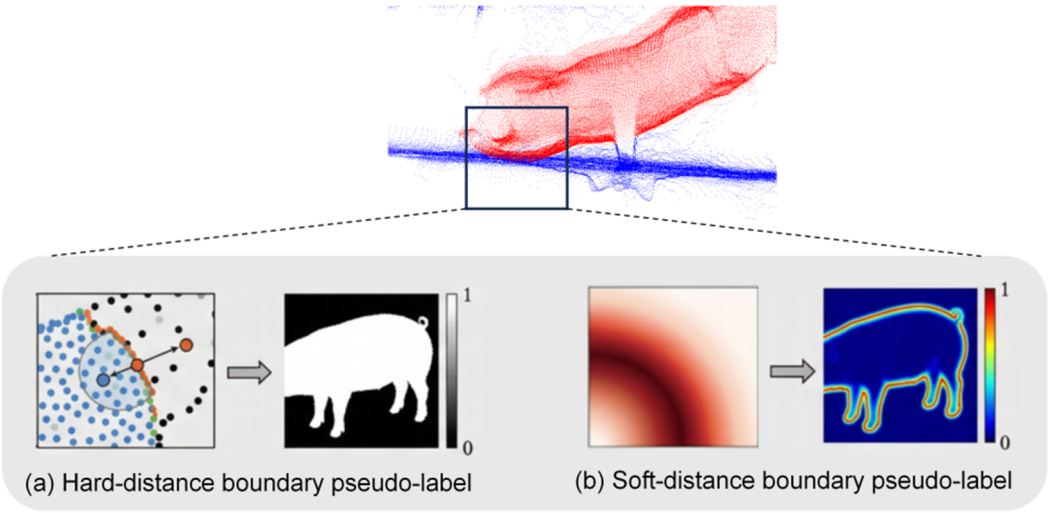

Figure 6. Illustration of the soft-distance boundary pseudo-labeling strategy. In subfigure (**a**), the blue and black dots represent pig and background points, respectively. The enlarged blue dot denotes the query point, while the orange dots indicate the nearest out-of-class points used to calculate the cross-class distance *di*. The green dots highlight boundary candidates near the pig–background interface, and the gray circle represents the local search neighborhood with radius *r*. In subfigure (b), the hard boundary pseudo-labels are converted into continuous soft-distance pseudo-labels according to the distance from the boundary. Warmer colors indicate higher pseudo-label values for points closer to the boundary, whereas cooler colors indicate lower values for points farther from the boundary.

2.3.4. Bidirectional Cross Boundary–Semantic Enhancement Module

Relying solely on multi-scale feature fusion remains insufficient to fully resolve the semantic adhesion problem in boundary regions. Existing boundary enhancement methods typically leverage boundary features to guide semantic attention computation [18]; however, boundary structural information does not fully participate in updating semantic features. When boundary prediction results are unstable or multi-scale ambiguities exist, unidirectional information flow is insufficient to effectively mitigate category confusion in contact regions between the pig body and the background.

To enhance the interaction between semantic features and boundary features, this study designs a bidirectional cross-boundary semantic module. This module first separates the multi-scale fused features into semantic and boundary features, and then facilitates explicit interaction between them through bidirectional cross-attention. Unlike unidirectional boundary guidance methods, boundary features in this module not only act as queries in attention computation but also participate as keys and values in updating semantic features, thereby improving the transfer of boundary structural information into the semantic space. The bidirectional cross-boundary semantic module is illustrated in Figure 7.

Specifically, given an input feature $f_M \in R^{N\times C}$, it is first decomposed into a semantic feature f_S and a boundary feature f_B through two independent mapping functions.

$$f_S = \phi_S(f_M),\ f_B = \phi_b(f_M) \tag{4}$$

In the formula, both $\phi_b(\cdot)$ and $\phi_S(\cdot)$ consist of multi-layer perceptrons. Subsequently, the queries, keys, and values required for multi-head attention are constructed as follows:

$$\begin{aligned} Q_S &= f_S W_q^S, \quad K_S = f_S W_k^S, \quad V_S = f_S W_v^S, \\ Q_b &= f_B W_q^b, \quad K_b = f_B W_k^b, \quad V_b = f_B W_v^b. \end{aligned} \tag{5}$$

On this basis, bidirectional cross-attention is applied to mutually enhance both semantic and boundary features.

$$\Delta f_S = \mathrm{Softmax}\left(\frac{\mathrm{Q_S} K_b^T}{\sqrt{d}}\right) V_b \tag{6}$$

$$\Delta f_B = \mathrm{Softmax}\left(\frac{\mathrm{Q_b} K_S^T}{\sqrt{d}}\right) V_S \tag{7}$$

In the formula, d denotes the feature dimension and is used to scale the inner product, thereby stabilizing the training process. To suppress interference from noisy boundary predictions on the semantic representation of non-boundary regions, a point-level gating vector g is further introduced:

$$g = \sigma(MLP([f_B, E])) \tag{8}$$

Among them, E denotes the boundary prediction probability, $[\cdot,\cdot]$ represents feature concatenation, and $\sigma(\cdot)$ is the Sigmoid function. Finally, the semantic and boundary features are updated as follows:

$$f_S^{'} = f_S + g \odot \Delta f_S \tag{9}$$

$$f_B^{'} = f_B + (1-g) \odot \Delta f_B \tag{10}$$

In the formula, $\odot$ denotes element-wise multiplication. This gating mechanism enables the model to adaptively regulate information flow according to point-wise boundary confidence. Specifically, when a point has a high boundary probability, the semantic branch more effectively incorporates boundary structural information; when a point is far from the boundary, the gating mechanism suppresses the interference of boundary information with semantic representation, while the boundary branch obtains a more stable structural prior from the semantic context. Through this mechanism, the model achieves stronger boundary discrimination capability in contact regions between the pig body and background structures such as the floor and railings.

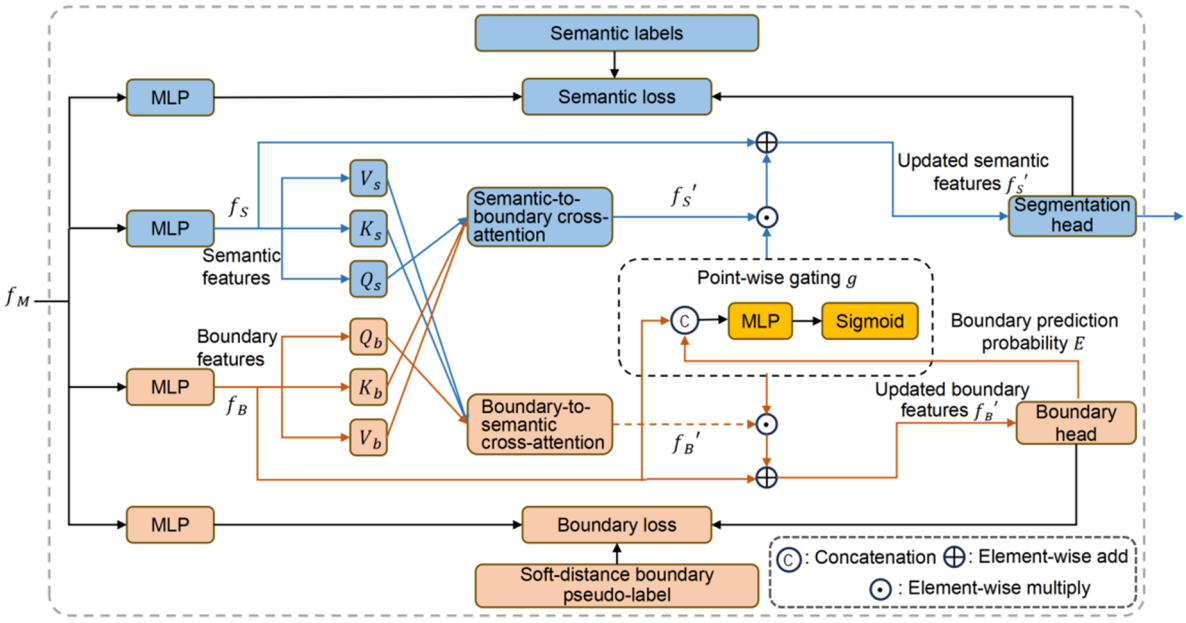

Figure 7. Architecture of the bidirectional cross boundary–semantic enhancement module. Semantic and boundary features are updated through bidirectional cross-attention and point-level gating, enabling adaptive information exchange between the semantic branch and the boundary branch.

2.3.5. Joint Loss Function

To simultaneously optimize pig semantic segmentation and boundary prediction, this study jointly employs cross-entropy (CE) loss [28], binary cross-entropy (BCE) loss, and Dice loss [29] to construct a joint loss function during training, thereby enabling supervised learning for both the semantic prediction branch and the boundary prediction branch. Given a predicted distribution X containing N points and the corresponding ground-truth label Y, Dice loss is defined as follows:

$$\mathrm{Dice}(X,Y)=1-\frac{2\sum_{i=1}^{N}X_i^TY_i}{\sum_{i=1}^{N}\left(X_i^TX_i+Y_i^TY_i\right)} \tag{11}$$

The semantic prediction score is denoted as $P \in [0, 1]^{N\times M}$, where N represents the number of points and M represents the number of semantic classes. To simultaneously account for point-wise classification accuracy and the consistency of overlapping predicted regions, cross-entropy loss and Dice loss are combined to construct the semantic segmentation loss L_{sem}.

$$L_{sem}=\frac{1}{N}\sum_{i=1}^{N}CE\left(P_i,P_i^g\right)+Dice\left(P_i,P_i^g\right) \tag{12}$$

In the formula, P_i denotes the semantic prediction probability of the *i-th* point, and P_i^g denotes its corresponding ground-truth semantic label. For the boundary prediction task, the network outputs the probability of each point being a boundary point, and the boundary score is denoted as $E \in [0, 1]^{N\times 1}$, where E_i represents the boundary prediction probability of the *i-th* point. The boundary loss L_{bou} is composed of binary cross-entropy loss and Dice loss:

$$L_{bou}=\frac{1}{N}\sum_{i=1}^{N}BCE\left(E_i,E_i^g\right)+Dice\left(E_i,E_i^g\right) \tag{13}$$

In the formula, E_i^g denotes the soft distance boundary pseudo-label corresponding to the *i-th* point. Finally, the overall loss function is defined as follows:

$$L=L_{sem}+\lambda L_{bou} \tag{14}$$

In Equation (14), λ balances the semantic segmentation loss and the boundary prediction loss. In the present implementation, λ was set to 1.0, assigning equal nominal weight to the semantic and boundary loss terms in the joint objective. This value was fixed in the implementation configuration before the final test evaluation and was used consistently in all reported experiments. It was not obtained through an exhaustive grid search or formal validation-set optimization. The sensitivity of the results to λ remains to be investigated in future work.

2.4. Experimental Setup and Evaluation Metrics

The training, validation, and test subsets were used for parameter estimation, model selection, and final evaluation, respectively. Training was conducted on an NVIDIA RTX A6000 GPU (NVIDIA Corporation, Santa Clara, CA, USA) with an Intel Xeon Platinum 8375C (Intel Corporation, Santa Clara, CA, USA) CPU, 128 GB RAM, Ubuntu 20.04.6 LTS, CUDA 12.4, and PyTorch 2.5.0. The proposed framework was implemented within the Pointcept v1.6.0 codebase. The main hardware and software configuration is summarized in Table 2.

Table 2. Experimental environment configuration.

Development Environment	Configuration
Processor	Intel(R) Xeon(R) Platinum 8375C CPU@2.90 GHz
GPU	NVIDIA RTX A6000 with 48 GB memory
RAM	128 GB
Operating system	Ubuntu 20.04.6 LTS
CUDA	12.4
PyTorch	2.5.0
Point-cloud framework	Pointcept v1.6.0

The training configuration used a per-GPU batch size of 1 and four-step gradient accumulation, corresponding to an effective batch size of four samples per optimization step. The model was trained for 100 epochs using the AdamW optimizer with an initial learning rate of 0.001 and a weight decay of 0.05. The learning rate was warmed up during the initial training stage and reduced by a factor of 0.1 at 60% and 90% of the total training schedule. Eight data-loading workers were used, and automatic mixed-precision training with bfloat16 was enabled. Model selection was performed according to the validation-set *mIoU*, and the test set was used only for final evaluation.

The present evaluation was designed as a controlled application-oriented comparison. Its primary purpose was to examine the behavior of the complete segmentation workflow under a fixed and independent animal-level split. Repeated-seed experiments would address the complementary question of variability across random

initializations and optimization trajectories. Accordingly, the current results are interpreted as descriptive evidence under the specified evaluation protocol rather than as inferential estimates of population-level performance.

The OctFormer backbone was configured with an octree depth of 11, a full depth of 2, an octree scale factor of 2048, and an FPN feature width of 168. The stage channels were set to (96, 192, 384, 384), the numbers of blocks were (2, 2, 18, 2), and the attention heads were (6, 12, 24, 24). The patch size was 26, the dilation factor was 4, and the drop-path rate was 0.5.

The preprocessing and training-only augmentation procedures described in Section 2.2 were applied consistently to all evaluated models. The same fixed physical ROI rule was applied to all samples before animal-level dataset partitioning.

To comprehensively evaluate pig point-cloud segmentation, this study used accuracy (*Acc*), mean intersection over union (*mIoU*), and boundary intersection over union (*B-IoU*) [19]. Acc measures the overall classification correctness of all points, whereas *mIoU* evaluates the average overlap between predicted regions and ground-truth labels across the two classes. Because overall *Acc* and *mIoU* may obscure localized contact errors, *B-IoU* was used as a complementary boundary metric for assessing segmentation quality near contacts between the pig body and background structures such as the floor, fences, and railings. Accuracy (*Acc*) is defined as follows:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

where *TP* denotes the number of pig-body points correctly classified as pig body, *TN* denotes the number of background points correctly classified as background, *FP* denotes the number of background points incorrectly classified as pig body, and *FN* denotes the number of pig body points incorrectly classified as background. For category *C*, the intersection over union is defined as:

$$IoU_c = \frac{\left|\text{Prediction}_c \bigcap \text{GroundTruth}_c\right|}{\left|\text{Prediction}_c \bigcup \text{GroundTruth}_c\right|} \tag{16}$$

In this binary classification task, the two categories correspond to the pig body and background. Therefore, mean Intersection over Union (*mIoU*) is defined as:

$$mIoU = \frac{1}{|C|}\sum_{c=1}^{C} IoU_c \tag{17}$$

Because *mIoU* is calculated as the unweighted mean of the pig-body *IoU* and background *IoU*, the two classes contribute equally to this metric irrespective of their relative point frequencies. Therefore, *mIoU* was reported together with *Acc* to reduce the influence of class-frequency imbalance on performance interpretation.

Since this study focuses on boundary segmentation in contact regions between pigs and background structures such as railings and floors within complex pigsty environments, accuracy (*Acc*) and mean Intersection over Union (*mIoU*) alone may not fully capture fine-grained missegmentation at boundaries. Therefore, *B-IoU* is additionally adopted to evaluate the model's boundary recognition capability. First, the sets of ground-truth boundary points B_l and predicted boundary points B_p are determined based on the neighborhood relationships of points: a point is considered a boundary point if its neighborhood contains points of different classes. Following previous studies, the neighborhood radius was set to r = 100 mm (0.1 m) [11,20] in the millimetre-based coordinate system used for the saved point clouds. *B-IoU* is defined as follows:

$$B\text{-}IoU = \frac{\left|B_l \bigcap B_p\right|}{\left|B_l \bigcup B_p\right|} \tag{18}$$

The same procedure was applied to every evaluated model. Given the conceptual relationship between the distance-based SDBPL objective and the spatial-proximity definition of *B-IoU*, *B-IoU* was interpreted as complementary boundary evidence rather than as independent confirmation of a broad performance advantage. The boundary results were therefore considered together with *Acc*, *mIoU*, ablation results, and qualitative comparisons under the current evaluation protocol. Boundary precision and recall were not included in the present evaluation. Accordingly, *B-IoU* was interpreted jointly with the qualitative comparisons, where retained background structures and omitted pig-body edges were examined as manifestations of over-segmentation and under-segmentation, respectively. The neighborhood radius used for SDBPL and the spatial radius used to define and dilate boundary regions for *B-IoU* were identical in the implementation: r = 100 coordinate units (100 mm, 0.1 m).

3. Results and Analysis

3.1. Experimental Design and Comparative Methods

To evaluate the proposed method, all models were trained and tested using the same prespecified animal-level split and the same reported metrics. This split was constructed within the available two-farm dataset and was not designed as a cross-farm generalization experiment. In particular, no model was trained exclusively on one farm and evaluated exclusively on the other farm. Therefore, the comparative results quantify performance under the sampled acquisition distribution rather than robustness to an unseen farm, housing configuration, or camera deployment. The reported *Acc*, *mIoU*, and *B-IoU* values are descriptive point estimates on the prespecified independent animal-level test split. No null-hypothesis significance tests were performed; consequently, no *p*-values or multiple-comparison correction were applied. The three metrics characterize different aspects of the same segmentation outputs and were not treated as separate confirmatory hypotheses.

To ensure a fair experimental comparison, this study selected five representative point cloud semantic segmentation models as comparative methods: PointNet++, OctFormer, OA-CNNs, PointStack, and Point Transformer V3 (PTv3). Among them, PointNet++ is a classic hierarchical point cloud network that performs semantic segmentation on unordered point sets through sampling, grouping, and local feature aggregation, serving as a fundamental baseline for point cloud segmentation [9]. OctFormer organizes point clouds using an octree structure and performs Transformer-based attention computation within local windows, thereby enhancing point cloud feature representation while reducing computational complexity [14]. OA-CNNs improve the modeling of local structures in sparse point clouds through adaptive receptive fields and adaptive relationship convolution [30]. PointStack aggregates point features from multiple layers at different resolutions while preserving both high-level semantic information and high-resolution details [31]. PTv3 expands the receptive field by serializing point clouds and simplifying the Transformer structure, thereby improving the efficiency of point cloud feature computation [32].

The comparison models encompass diverse technical approaches, including hierarchical point networks, octree transformers, convolutional point-cloud networks, multi-resolution feature aggregation networks, and serialized transformers. To support a consistent comparison, all models were evaluated using the same prespecified animal-level training, validation, and test split, the same binary pig-background label definition, and the same *Acc*, *mIoU*, and *B-IoU* evaluation procedures. The validation set was used for model selection, while the test set was reserved exclusively for final evaluation. Because these models employ different point-sampling, convolution, serialization, and octree-based processing strategies, their architecture-specific input organization and hyperparameters followed the corresponding implementations and recommended settings. No model-specific adjustment was performed using the test-set results. Accordingly, the comparison represents the observed segmentation performance of the evaluated implementations under a common dataset partition and evaluation protocol.

The reported comparative metrics were calculated on the pooled test split, which included the available samples from both the three-view and seven-view acquisition configurations. The pooled evaluation was used to assess a single segmentation model across the two practical acquisition settings; acquisition-specific metrics were not calculated separately in the current evaluation.

The purpose of this comparison was to examine whether the integrated boundary-aware workflow provides consistent performance differences under a common evaluation protocol. The reported *Acc*, *mIoU*, and *B-IoU* values are descriptive point estimates from the final runs. No null-hypothesis significance tests were performed; consequently, no *p*-values or multiple-comparison correction were applied. Repeated-seed training, repeated animal-level splits, and confidence-interval estimation were not part of the present evaluation design. The results should therefore be interpreted as evaluation-specific empirical observations rather than as statistically confirmed or universally generalizable improvements.

3.2. Comparative Experiments on Point Cloud Segmentation

The quantitative evaluation results of all models on the prespecified independent animal-level test split are presented in Table 3. Among the evaluated runs, the proposed method produced the highest observed *Acc*, *mIoU*, and *B-IoU* values of 98.65%, 97.36%, and 88.77%, respectively. Relative to OctFormer, the observed differences were 0.55, 0.35, and 3.37 percentage points, respectively. The *Acc* and *mIoU* margins were modest, whereas the larger *B-IoU* difference was consistent with the explicit boundary-supervision design of the proposed framework. These differences are interpreted descriptively under the current test-set protocol rather than as evidence of statistical significance. Moreover, because SDBPL and *B-IoU* both characterize boundary regions through spatial proximity, the *B-IoU* result is treated as complementary evidence and is interpreted together with *Acc*, *mIoU*, the ablation results, and the qualitative analysis.

Table 3. Comparative experimental results for pig point-cloud segmentation.

Model	*Acc* (%) ↑	*mIoU* (%) ↑	*B-IoU* (%) ↑
PointNet++	95.10	93.89	70.20
OctFormer	98.10	97.01	85.40
OA-CNNs	97.98	96.10	80.89
PointStack	96.45	95.02	76.09
PTv3	98.02	96.89	83.10
Proposed method	98.65	97.36	88.77

Note: *Acc*, accuracy; *mIoU*, mean Intersection over Union; *B-IoU*, boundary Intersection over Union. ↑ indicates that a higher value represents better performance. The reported values are descriptive point estimates from the corresponding final runs on the prespecified test split. Variability across random initializations was not assessed.

Among the comparison models, PointNet++ achieves *Acc* and *mIoU* values of 95.10% and 93.89%, respectively, whereas its *B-IoU* is only 70.20%. This indicates that PointNet++ can perform basic binary segmentation of the pig body and background but has limited capability in recognizing complex boundary regions. This result is closely related to its network architecture: PointNet++ primarily relies on hierarchical sampling, local neighborhood grouping, and feature aggregation for point cloud representation, which limits its ability to model fine structural variations and category transition relationships at contact regions between the pig body, railings, and the floor. Consequently, it is more prone to misclassifying background points as pig body points or missing pig body edges in boundary adhesion regions.

OctFormer organizes point clouds using an octree structure and performs self-attention computations within local windows. Its overall segmentation performance surpasses that of PointNet++, with *Acc* and *mIoU* reaching 98.10% and 97.01%, respectively. However, OctFormer's *B-IoU* is 85.40%, still lower than that achieved by the proposed method. This observed pattern suggests that a general octree Transformer backbone improves overall semantic representation, while additional boundary-specific modeling may be beneficial for contact regions.

OA-CNNs, PointStack, and PTv3 enhance point cloud feature representation from the perspectives of adaptive convolution, multi-resolution feature aggregation, and serialized Transformer, respectively. Among these models, OA-CNNs achieve a *B-IoU* of 80.89%, indicating that adaptive receptive fields and local geometric modeling contribute to boundary recognition. PTv3 attains *Acc* and *mIoU* values of 98.02% and 96.89%, respectively, with a *B-IoU* of 83.10%, suggesting that serialized Transformers improve context modeling but still lag behind the proposed method in boundary segmentation quality. PointStack achieves a *B-IoU* of 76.09%, highlighting that relying solely on multi-resolution feature stacking is insufficient to fully address complex boundary adhesion at contact regions between the pig body and railings or the floor.

Overall, the comparative results indicate that the boundary-specific components primarily improve segmentation near pig-background contact regions while maintaining the overall semantic segmentation performance. The performance comparison across the evaluated models is summarized in Figure 8.

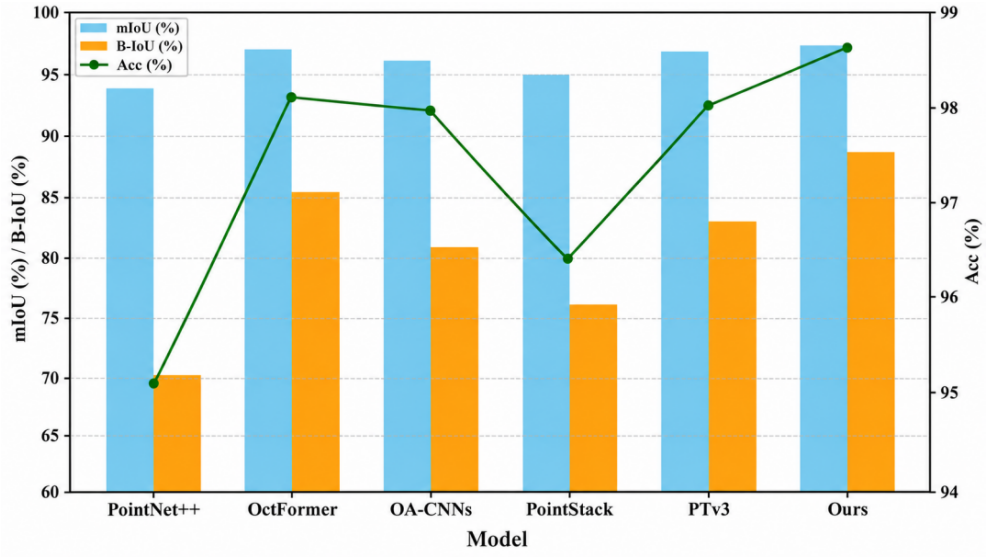

Figure 8. Comparative performance of different point cloud segmentation models. *Acc*, accuracy; *mIoU*, mean Intersection over Union; *B-IoU*, boundary Intersection over Union. Higher values indicate better segmentation performance.

3.3. Visual Analysis of Segmentation Results

To further analyze segmentation differences among models in complex pigsty scenarios, this study selects representative pig body point cloud samples from the test set for visual comparison, as shown in Figure 9. Overall, all models can identify the main body of the pig; however, clear qualitative differences are visible in segmentation quality at contact regions between the pig body and background structures such as railings and the floor. PointNet++ and PointStack are prone to misclassifying background points as pig body points, erroneously deleting pig body edges, or producing discontinuous local boundaries in areas where the pig's head is close to a railing, the side of the torso overlaps with a fence, or the legs are near the floor. These errors manifest as boundary adhesion, contour loss, and incorrect category attribution. Compared with PointNet++, OA-CNNs and PTv3 show some improvement in boundary regions; nevertheless, when there are large variations in point cloud density across adjacent areas or when the pig body is close to background structures, local mis-segmentation still occurs.

In contrast, the proposed method produces clearer pig body contours and less background residue in these challenging areas. In regions where the pig's head contacts a railing, the method more effectively distinguishes slender railing points from pig body points, reducing misclassification of railing points as part of the pig body. In areas where the legs meet the floor, the model better preserves limb edges and reduces mis-segmentation caused by adhesion of floor points. In regions where the side of the torso overlaps with a fence, boundary continuity is improved, and the outer contour of the pig body is more complete. These qualitative patterns are consistent with the observed *B-IoU* difference in Table 3 and suggest that the main observed benefit of the proposed workflow occurs in complex contact-boundary regions rather than in the classification of the main pig-body region.

The enlarged comparisons in Figure 9 also show that the proposed method does not eliminate all local boundary errors. Small amounts of background residue and minor pig-body edge omission remain in some circled contact regions, particularly where railing points are spatially close to the pig surface or where limb points near the floor have relatively sparse local coverage. These residual errors indicate that highly similar local geometry and reduced point density remain challenging for fine pig-background separation.

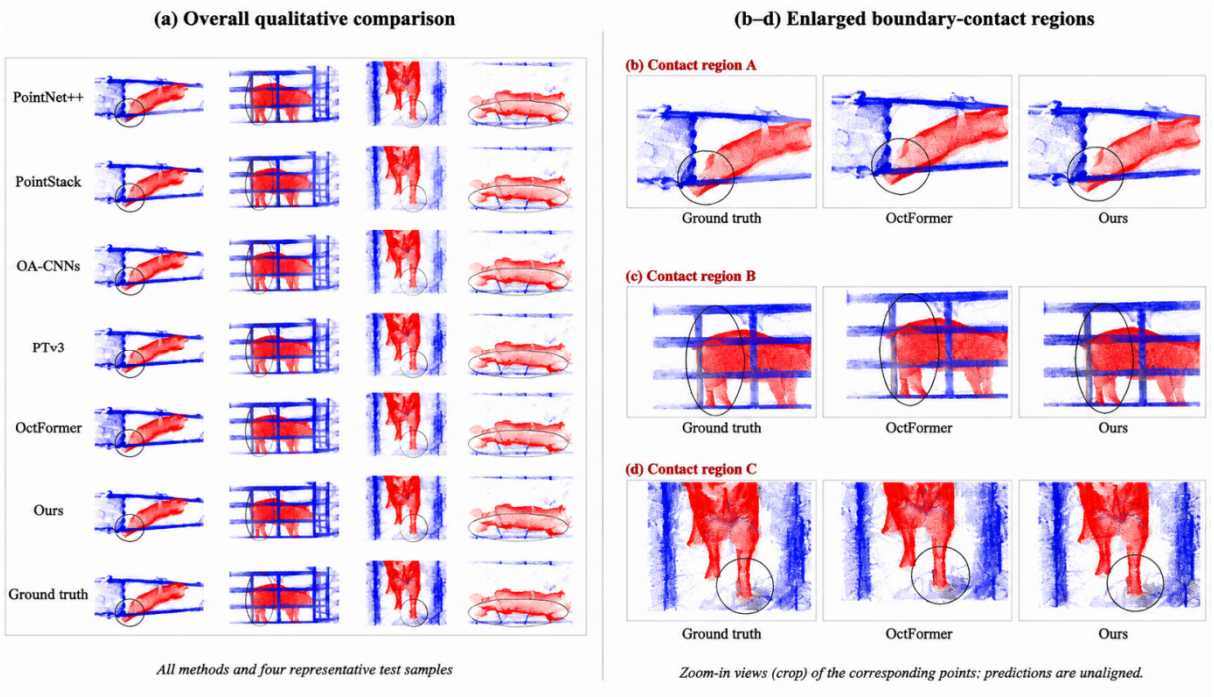

Figure 9. Qualitative comparison of pig point-cloud segmentation in representative boundary-contact regions. (**a**) Overall qualitative comparison of different methods on four representative test samples. (**b**) Enlarged view of contact region A, showing the head–railing contact region. (**c**) Enlarged view of contact region B, showing the torso–fence contact region. (**d**) Enlarged view of contact region C, showing the leg–floor contact region. The ground truth, OctFormer prediction, and proposed-method prediction are presented for comparison in the enlarged views. Circles indicate representative retained background points and omitted pig-body boundary points.

This phenomenon can be attributed to the introduction of soft distance boundary pseudo-labels and the bidirectional cross-boundary semantic module on the basis of multi-scale feature representation. The former mitigates training instability caused by abrupt changes in hard labels near boundaries through continuous boundary

supervision, whereas the latter facilitates information interaction between boundary features and semantic features, enabling the model to extract more discriminative contact-interface features. Under the evaluated conditions, these mechanisms are consistent with the observed reduction in boundary errors associated with uneven point density, local occlusion, and background adhesion.

3.4. Ablation Study on Network Modules

To evaluate the contributions of each key module to model performance, this study conducts a module ablation experiment. The results are summarized in Table 4. Model A serves as the baseline, using OctFormer as the backbone and incorporating the original boundary semantic module. In this baseline structure, the boundary branch primarily functions as a query to guide semantic attention computation, and boundary information is not fully integrated into the semantic feature update. Model B adds the Bidirectional Cross-Boundary Attention (BCBA) module to the baseline, enhancing bidirectional information interaction between semantic and boundary features. Model C incorporates the Soft Distance Boundary Pseudo-Label (SDBPL) into the baseline, providing continuous boundary supervision. The final model combines both BCBA and SDBPL.

Table 4. Ablation study results for network modules.

Model	Baseline	BCBA	SDBPL	*Acc* (%) ↑	*mIoU* (%) ↑	*B-IoU* (%) ↑
A	√			98.15	97.06	85.98
B	√	√		98.53	97.28	88.02
C	√		√	98.22	97.13	86.98
Proposed	√	√	√	98.65	97.36	88.77

Note: BCBA, Bidirectional Cross-Boundary Attention; SDBPL, Soft Distance Boundary Pseudo-Label; *Acc*, accuracy; *mIoU*, mean Intersection over Union; *B-IoU*, boundary Intersection over Union. √ indicates that the corresponding module was included. ↑ indicates that a higher value represents better performance. All values are descriptive point estimates from single runs under the same data split. The observed differences were not tested for statistical significance.

As shown in Table 4, the baseline model (Model A) already achieves high overall segmentation performance, with *Acc* and *mIoU* reaching 98.15% and 97.06%, respectively. However, its *B-IoU* is 85.98%, indicating that relying solely on the OctFormer backbone and the unidirectional boundary guidance mechanism remains insufficient to fully resolve boundary confusion in complex contact regions. These results further suggest that, even when overall semantic segmentation accuracy is high, errors in boundary regions continue to be the primary factor limiting further improvement of the model.

After incorporating BCBA into the baseline model, Model B achieves *Acc*, *mIoU*, and *B-IoU* values of 98.53%, 97.28%, and 88.02%, respectively. Notably, its *B-IoU* is 2.04 percentage points higher than that of Model A, representing a greater improvement compared with *Acc* and *mIoU*. This observed result is consistent with the intended role of bidirectional interaction between semantic and boundary features in representing contact boundaries. Unlike unidirectional boundary guidance, BCBA allows both boundary and semantic features to participate in information updates, facilitating the transfer of fine-grained structural information at junctions between the pig body, railings, and the floor to the semantic branch, thereby improving category discrimination in boundary regions.

After adding SDBPL, Model C achieved *Acc*, *mIoU*, and *B-IoU* values of 98.22%, 97.13%, and 86.98%, respectively. Relative to Model A, the observed *B-IoU* difference was +1.00 percentage point. This pattern is consistent with the intended role of soft-distance boundary supervision in providing a continuous training signal near category-transition regions. However, because the ablation results were obtained from single runs, this difference is interpreted as descriptive evidence rather than as a statistically confirmed independent effect of SDBPL. Compared with BCBA, the observed contribution of SDBPL alone was relatively modest under the current evaluation protocol.

When BCBA and SDBPL are applied simultaneously, the proposed method achieved the highest observed values among the evaluated configurations, with *Acc*, *mIoU*, and *B-IoU* reaching 98.65%, 97.36%, and 88.77%, respectively. Compared with the baseline model (Model A), these three metrics improve by 0.50, 0.30, and 2.79 percentage points, respectively. Relative to models using only BCBA or SDBPL, the proposed method achieved the highest observed values across the reported metrics. These results are consistent with complementary roles of BCBA and SDBPL. BCBA enhances interaction between boundary and semantic information at the network structure level, while SDBPL mitigates training uncertainty caused by boundary blurring and point cloud density variations at the supervision signal level. The joint use of these modules is consistent with improved boundary

localization under the current evaluation protocol, although the contribution of each component should be interpreted together with the limitations of the single-run ablation results and the conceptual relationship between SDBPL and *B-IoU*.

The ablation results provide additional, although not independent, support for the proposed design. Adding BCBA to the baseline improved *B-IoU* from 85.98% to 88.02%, while adding SDBPL alone increased *B-IoU* to 86.98%. The complete model combining BCBA and SDBPL achieved the highest *Acc*, *mIoU*, and *B-IoU* values. These observations are consistent with the intended contribution of the two modules, but they do not establish statistical significance or fully separate the effect of distance-based supervision from the potential alignment between SDBPL and *B-IoU*.

3.5. Ablation Study on Loss Functions

To further analyze the impact of loss function design on model performance, this study conducts ablation experiments on loss functions using the network with both BCBA and SDBPL incorporated. For the semantic branch, either cross-entropy (CE) loss or a combined CE + Dice loss is used. For the boundary branch, either binary cross-entropy (BCE) loss or a combined BCE + Dice loss is used. The experimental results are summarized in Table 5.

Table 5. Ablation study results for loss functions.

No.	Semantic Branch	Boundary Branch	*Acc* (%) ↑	*mIoU* (%) ↑	*B-IoU* (%) ↑
1	CE	BCE	98.48	97.24	87.69
2	CE	BCE + Dice	98.53	97.29	88.33
3	CE + Dice	BCE	98.54	97.27	87.98
4	CE + Dice	BCE + Dice	98.65	97.36	88.77

Note: CE, cross-entropy; BCE, binary cross-entropy; Dice, Dice loss; *Acc*, accuracy; *mIoU*, mean Intersection over Union; *B-IoU*, boundary Intersection over Union. ↑ indicates that a higher value represents better performance. All values are descriptive point estimates from single runs under the same data split. Variability across random initializations was not assessed.

In Experiment 1, only CE and BCE losses were used as the baseline configuration, resulting in *Acc*, *mIoU*, and *B-IoU* values of 98.48%, 97.24%, and 87.69%, respectively. In Experiment 2, after introducing Dice loss to the boundary branch, *Acc*, *mIoU*, and *B-IoU* increased to 98.53%, 97.29%, and 88.33%, representing gains of 0.05, 0.05, and 0.64 percentage points compared with Experiment 1. Among these metrics, the improvement in *B-IoU* was the most pronounced, indicating that Dice loss strengthens the regional overlap constraint between boundary predictions and soft boundary pseudo-labels. Because boundary points constitute a relatively small proportion of the overall point cloud, relying solely on BCE is susceptible to class imbalance. Dice loss optimizes the prediction region from the perspective of set overlap, providing stronger constraints on minority boundary regions and thereby improving boundary segmentation performance.

Experiment 3 adopted the CE + Dice combined loss in the semantic branch, while the boundary branch still used BCE. Compared with Experiment 1, *Acc*, *mIoU*, and *B-IoU* increased by 0.06, 0.03, and 0.29 percentage points respectively, indicating that introducing the Dice loss in the semantic branch helps improve the overall consistency between the predicted and true labels of the main region, but the improvement of the boundary metrics is less than that when the Dice loss is introduced in the boundary branch. Under the current evaluation protocol, this pattern suggests that introducing an overlap-based constraint in the boundary branch may be more relevant to B-IoU than introducing it only in the semantic branch.

When the combined CE + Dice loss is applied to both the semantic and boundary branches, Experiment 4 achieved the highest observed values among the evaluated loss configurations, with *Acc*, *mIoU*, and *B-IoU* reaching 98.65%, 97.36%, and 88.77%, respectively. Compared with Experiment 1, these three metrics increase by 0.17, 0.12, and 1.08 percentage points, respectively. Relative to Experiments 2 and 3, Experiment 4 also produced the highest observed values across the three reported metrics. These observations are consistent with complementary roles of point-wise classification supervision and regional-overlap constraints within the evaluated workflow. However, the relatively small differences among the loss configurations are interpreted descriptively and do not establish statistical significance or robustness across random initializations.

Taken together, the observed ablation results are consistent with contributions from octree-based multi-scale feature extraction, boundary supervision, semantic-boundary interaction, and loss-function design. Specifically, the observed results are consistent with the intended roles of SDBPL in providing continuous supervision near contact boundaries, BCBA in facilitating bidirectional interaction between boundary and semantic features, and

Dice loss in supporting overlap-based optimization for minority boundary regions. The configuration combining these components achieved the highest observed *Acc*, *mIoU*, and *B-IoU* values among the evaluated configurations. Because these ablation results were obtained from single runs, they should be interpreted as descriptive evidence consistent with the proposed design rather than as statistically established causal effects.

4. Discussion

On the prespecified test split, the proposed method produced the highest observed *Acc*, *mIoU*, and *B-IoU* among the compared runs. The *Acc* and *mIoU* margins were small, whereas the *B-IoU* margin was larger. These results are descriptive because the current study did not quantify variability across random initializations. Moreover, *B-IoU* is conceptually aligned with the distance-based SDBPL objective, while the general baselines do not receive equivalent boundary supervision. This alignment may favor boundary-aware methods; *B-IoU* is therefore interpreted jointly with *Acc*, *mIoU*, ablation results, and qualitative comparisons rather than as independent proof of broad superiority. A direct boundary-aware baseline would provide a stronger controlled comparison, but such a matched comparison was not included in the present study.

The distinction between descriptive comparison and inferential validation is important for interpreting the present study. Repeated-seed analysis would quantify variability associated with random initialization and optimization, but it addresses an uncertainty question that is complementary to the primary evaluation of the proposed workflow. The current experiment examines whether the integrated boundary supervision, semantic-boundary interaction, and loss design exhibit coherent quantitative and qualitative behavior under a fixed independent protocol. The available evidence is therefore used to characterize the workflow under the evaluated conditions, but not to claim statistical significance or universal superiority.

From a structural perspective, the effectiveness of the proposed method primarily stems from the hierarchical organization capability of the octree Transformer for sparse 3D point clouds. Pig point clouds are characterized by disorder, sparsity, and uneven local density, with pig bodies, railings, the floor, and noise points often interlaced in space. When relying solely on feature aggregation of unstructured point sets, the model is prone to misclassification in regions with local density variations or background adhesion. The octree structure organizes point clouds into non-empty nodes with spatial hierarchical relationships and performs attention computations within local windows, reducing computational burden while preserving local geometric relationships. Further, multi-scale feature fusion allows the model to simultaneously leverage high-level semantic information and low-level boundary details, avoiding problems such as overly thick boundaries or local structure fragmentation caused by using single-scale features alone.

The key distinction between the proposed framework and general point-cloud semantic-segmentation networks lies in its explicit modeling of ambiguous contact boundaries. The framework uses the soft-distance boundary pseudo-label to provide continuous distance-based supervision and employs the bidirectional cross-boundary semantic module to promote interaction between semantic and boundary features. This design complements the multi-scale representation of the Octree Transformer and is intended to improve feature discrimination in locally ambiguous regions. The module-level contributions are evaluated descriptively through the ablation experiments presented in Section 3.4.

The ablation results are consistent with complementary roles of BCBA and SDBPL. BCBA facilitates bidirectional interaction between semantic and boundary features, whereas SDBPL provides continuous supervision near category-transition regions. Their combination produced the highest observed values among the evaluated configurations. Because the ablation results were obtained from single runs, they are interpreted as descriptive evidence rather than as evidence of statistical significance.

Reliable foreground segmentation is an upstream requirement for point-cloud completion, keypoint localization, body-size measurement, and body-mass modeling. Retained railing or floor points can introduce false structures, whereas removal of limb, abdominal, or dorsal edges can alter cross-sectional measurements. Related work in typical swine pens has shown how point-cloud geometry supports body-mass prediction pipelines [23], further motivating careful boundary evaluation. The present study, however, did not directly quantify downstream error reduction. Therefore, the potential benefits for completion and phenotyping are presented as application motivation rather than as experimentally verified downstream improvements.

Several limitations constrain the interpretation of the present results. First, the reported values are single-run point estimates on the prespecified independent animal-level test split. Training variability across random initializations, confidence intervals, repeated animal-level splits, and formal statistical comparisons were not assessed. Consequently, the robustness of the small *Acc*, *mIoU*, and ablation margins to random initialization,

optimization stochasticity, and repeated training has not been established. The observed differences should therefore be interpreted as evaluation-specific margins rather than as evidence of statistical significance.

Second, cross-farm and cross-condition generalization remains an important limitation. The dataset was collected from two farms with relatively stable camera heights, passage widths, railing structures, and acquisition procedures. Although the animal-level split prevents samples from the same pig from appearing in different subsets, the available records do not preserve a verified farm-by-split allocation table, and the original protocol did not include leave-one-farm-out validation. Consequently, farm-specific background cues, including flooring appearance, railing morphology, illumination, and housing structure, may be present in both the training and test distributions. The current animal-level split provides an independent evaluation across the sampled pigs while preventing identity-related data leakage. Because the dataset includes two commercial farms and both three-view and seven-view acquisition configurations, it represents multiple practical acquisition conditions. Severe motion blur, jumping, and incomplete acquisitions were excluded during quality screening; therefore, the present evaluation focuses on usable point-cloud captures obtained during routine passage through the acquisition system. Future farm-held-out and subgroup analyses will further characterize the transferability of the method across farms, acquisition configurations, body-weight ranges, boundary-contact scenarios, and more challenging conditions involving severe occlusion, unusual postures, or low point density.

Third, all samples were annotated by a single trained annotator using a common annotation protocol. This procedure improves procedural consistency and avoids variability caused by different annotation habits, but it does not provide a quantitative estimate of inter-annotator agreement or annotation repeatability. This issue is particularly relevant at pig-floor, pig-fence, and pig-railing contact regions, where occlusion, sensor noise, point-density variation, and physical contact may make the exact boundary subject to reasonable annotation judgment. Because the same labels were used to generate the SDBPL targets and to define the reference boundaries for *B-IoU*, annotation uncertainty may influence both boundary supervision and boundary evaluation. The reported boundary results are therefore interpreted as performance under the current annotation protocol rather than as an annotation-independent estimate of boundary accuracy. Independent double annotation, repeat annotation, and agreement analysis should be conducted in future work.

Fourth, the evaluation has a potential coupling between the SDBPL training objective and the *B-IoU* evaluation metric. Both characterize boundary regions using spatial proximity, although they are applied at different stages of the pipeline. Therefore, the observed *B-IoU* improvement may partly reflect the alignment between the training objective and the evaluation definition. None of the comparison models used an auxiliary boundary-supervision objective equivalent to SDBPL. A more controlled assessment would require a boundary-aware OctFormer baseline using an auxiliary hard-boundary loss without SDBPL or the bidirectional cross-boundary semantic module. Such a matched baseline was not available in the current study. Consequently, the contributions of SDBPL and the bidirectional semantic-boundary interaction should be interpreted together with *Acc*, *mIoU*, ablation results, and qualitative evidence rather than as independently established causal effects.

Finally, sensitivity to the SDBPL neighborhood radius, the transition-width parameter σ, the loss-balancing weight λ, point density, and camera layout was not systematically evaluated. The current parameter values were fixed in the implementation configuration and were not supported by an exhaustive grid search or formal sensitivity analysis. Overfitting was mitigated through animal-level separation, geometric augmentation, voxel subsampling, stochastic point dropout and cropping, AdamW weight decay, stochastic depth, scheduled learning-rate decay, and validation-set model selection. The independent test set was not used for hyperparameter selection. These measures reduce but do not eliminate overfitting risk, particularly in the absence of external-farm validation. The proposed model contains approximately 44.28 million parameters, compared with approximately 44.00 million parameters for the OctFormer backbone implementation. Based on these rounded values, the boundary-enhancement components introduce an additional approximately 0.28 million parameters, corresponding to a relative increase of about 0.64%. This represents a modest parameter overhead relative to the backbone. Standardized per-point-cloud inference latency, throughput, and peak GPU memory usage were not recorded under a unified deployment benchmark, and training iteration time was not used as a substitute for inference latency. These deployment-oriented indicators will be evaluated under controlled hardware and input-size conditions in future work. Future work should therefore include repeated-seed training with the reporting of mean ± SD, confidence intervals, and appropriate statistical comparisons, together with farm-held-out validation, cross-device testing, independent annotation analysis, parameter-sensitivity studies, and deployment-oriented evaluation.

5. Conclusions

To address challenges in real pig-housing environments, including close contact, blurred boundaries, and local adhesion between pig point clouds and background structures such as the floor, fences, and railings, this study proposed a boundary-aware pig point-cloud segmentation method based on an Octree Transformer. The method integrates local geometric details and global semantic context through octree convolution, self-attention encoding, and multi-scale feature fusion. It further introduces soft-distance boundary pseudo-labels and a bidirectional cross-boundary semantic module to enhance boundary supervision and semantic-boundary feature interaction in pig-background contact regions.

On the prespecified independent animal-level test split, the proposed method yielded an Acc of 98.65%, an *mIoU* of 97.36%, and a *B-IoU* of 88.77%. Relative to OctFormer, the observed differences were 0.55, 0.35, and 3.37 percentage points, respectively. The *Acc* and *mIoU* margins were modest, whereas the larger *B-IoU* difference was consistent with the explicit boundary-oriented design. Because the results were obtained from single final runs, they are interpreted as descriptive observations rather than as evidence of statistical significance.

The contribution of this study lies in the integrated boundary-aware workflow and the coherent behavior observed across overall segmentation metrics, boundary-specific evaluation, module ablations, loss-function ablations, and qualitative contact-region analysis. The results support the applicability of the workflow under the sampled two-farm conditions and fixed acquisition configurations, but they do not establish robustness across random initializations or generalization to unseen farms, devices, or substantially different acquisition conditions. Future validation should include repeated-seed training with mean ± SD, confidence intervals, and appropriate statistical comparisons, as well as a matched boundary-aware baseline, independent annotation analysis, farm-held-out testing, cross-device evaluation, and deployment-cost analysis.

Supplementary Materials

The additional data and information can be downloaded at: https://media.sciltp.com/articles/others/2611161352545762/SAI-26060210-ARRIVE-checklist.pdf.

Author Contributions

Conceptualization, Z.X., F.S., Q.L. and W.M.; methodology, Z.X., F.S. and W.M.; software, Z.X., F.S. and X.Q.; validation, Z.X., F.S., Z.W. and M.G.; formal analysis, Z.X., F.S. and X.Q.; investigation, Z.X., Z.W., M.G. and Y.F.; resources, Q.L. and W.M.; data curation, Z.X., F.S., Z.W., M.G. and Y.F.; writing—original draft preparation, Z.X. and F.S.; writing—review and editing, X.Q., Z.W., M.G., Y.F., S.X.Y., Q.L. and W.M.; visualization, Z.X., F.S. and X.Q.; supervision, S.X.Y., Q.L. and W.M.; project administration, Q.L. and W.M.; funding acquisition, Q.L. and W.M.; All authors have read and agreed to the published version of the manuscript.

Funding

This research was funded by the Ability Innovation Program of Beijing Academy of Agriculture and Forestry Sciences (Grant No. KJCX20260922) and Beijing Digital Agriculture Innovation Consortium Project (BAIC10-2026).

Institutional Review Board Statement

This study involved non-invasive observational point-cloud data collection from pigs during routine commercial farm management. No invasive intervention, experimental treatment, alteration of routine management, or additional animal handling was involved. Data collection was conducted with permission from the participating farms. The study was conducted in accordance with the applicable animal husbandry requirements of the Animal Husbandry Law of the People's Republic of China (2022 revision), particularly Article 42, which requires livestock keepers to provide appropriate breeding conditions and living and growing environments for the animals under their care. No separate institutional animal ethics approval was obtained because the study involved only non-invasive observation and did not involve invasive procedures, experimental treatment, or additional animal handling. The study was reported in accordance with the ARRIVE 2.0 guidelines, as applicable to its non-invasive observational design. A completed ARRIVE 2.0 checklist is provided as Supplementary File S1.

Informed Consent Statement

Not applicable.

Data Availability Statement

The raw point-cloud datasets generated and analyzed during the current study are not publicly available because they contain farm-specific acquisition information and individual animal identification records, and their release is restricted by data-use and farm-management agreements. To facilitate independent assessment, anonymized representative point clouds and their corresponding segmentation labels may be made available from the corresponding author upon reasonable request and with permission from the participating farms. The complete source code and training scripts are not publicly available at this stage because they are part of an ongoing research project and are subject to institutional data-management and project-use restrictions. Model configuration files, evaluation scripts, relevant implementation details, and trained model parameters may be made available from the corresponding author upon reasonable request for academic and non-commercial research purposes, subject to applicable restrictions. All reported *Acc*, *mIoU*, and *B-IoU* values are descriptive test-set point estimates from the corresponding final runs. Repeated-seed training, confidence-interval estimation, repeated animal-level splits, cross-validation, and formal null-hypothesis significance tests were not performed. Consequently, no mean ± SD values, *p*-values, or multiple-comparison corrections are reported. The three metrics characterize complementary aspects of the same segmentation outputs and were not treated as separate confirmatory hypotheses. The numerical differences are therefore interpreted as evaluation-specific observations rather than as statistically confirmed improvements. The animal-level split prevented acquisitions from the same pig from crossing subsets, but it was not designed as an external-farm or repeated-split validation.

Conflicts of Interest

Given the role as Editor-in-Chief, Simon X. Yang had no involvement in the peer review of this paper and had no access to information regarding its peer-review process. Full responsibility for the editorial process of this paper was delegated to another editor of the journal. The other authors declare no conflicts of interest.

Use of AI and AI-Assisted Technologies

During the preparation of this work, the authors used ChatGPT (OpenAI) to assist with language editing, translation. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

1. Dohmen, R.; Catal, C.; Liu, Q. Computer vision-based weight estimation of livestock: A systematic literature review. *N. Z. J. Agric. Res.* **2022**, *65*, 227–247. https://doi.org/10.1080/00288233.2021.1876107.
2. Ma, W.; Qi, X.; Sun, Y.; et al. Computer Vision-Based Measurement Techniques for Livestock Body Dimension and Weight: A Review. *Agriculture* **2024**, *14*, 306. https://doi.org/10.3390/agriculture14020306.
3. Ma, W.; Guo, M.; Li, Z.; et al. Computer vision-based livestock weight measurement: Principles, methods, and challenges. *Intell. Robot.* **2026**, *6*, 205–228. https://doi.org/10.20517/ir.2026.11.
4. Wang, K.; Guo, H.; Ma, Q.; et al. A portable and automatic Xtion-based measurement system for pig body size. *Comput. Electron. Agric.* **2018**, *148*, 291–298. https://doi.org/10.1016/j.compag.2018.03.018.
5. Shi, S.; Yin, L.; Liang, S.; et al. Research on 3D surface reconstruction and body size measurement of pigs based on multi-view RGB-D cameras. *Comput. Electron. Agric.* **2020**, *175*, 105543. https://doi.org/10.1016/j.compag.2020.105543.
6. Guo, H.; Li, Z.; Ma, Q.; et al. A bilateral symmetry based pose normalization framework applied to livestock body measurement in point clouds. *Comput. Electron. Agric.* **2019**, *160*, 59–70. https://doi.org/10.1016/j.compag.2019.03.010.
7. Ruchay, A.; Kober, V.; Dorofeev, K.; et al. Accurate body measurement of live cattle using three depth cameras and non-rigid 3-D shape recovery. *Comput. Electron. Agric.* **2020**, *179*, 105821. https://doi.org/10.1016/j.compag.2020.105821.
8. Qi, C.R.; Su, H.; Mo, K.; et al. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 652–660. https://doi.org/10.1109/cvpr.2017.16.
9. Qi, C.R.; Yi, L.; Su, H.; et al. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5099–5108.
10. Wang, Y.; Sun, Y.; Liu, Z.; et al. Dynamic Graph CNN for Learning on Point Clouds. *ACM Trans. Graph.* **2019**, *38*, 1–12. https://doi.org/10.1145/3326362.
11. Thomas, H.; Qi, C.R.; Deschaud, J.E.; et al. KPConv: Flexible and Deformable Convolution for Point Clouds. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6410–6419. https://doi.org/10.1109/iccv.2019.00651.

12. Zhao, H.; Jiang, L.; Jia, J.; et al. Point Transformer. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 16239–16248. https://doi.org/10.1109/iccv48922.2021.01595.
13. Qian, G.; Li, Y.; Peng, H.; et al. PointNeXt: Revisiting PointNet++ with improved training and scaling strategies. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 23192–23204.
14. Wang, P.S. OctFormer: Octree-based Transformers for 3D Point Clouds. *ACM Trans. Graph.* **2023**, *42*, 1–11. https://doi.org/10.1145/3592131.
15. Lin, R.; Hu, H.; Wen, Z.; et al. Research on denoising and segmentation algorithm application of pigs' point cloud based on DBSCAN and PointNet. In Proceedings of the 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor), Trento-Bolzano, Italy, 3–5 November 2021; pp. 42–47. https://doi.org/10.1109/metroagrifor52389.2021.9628501.
16. Wang, S.; Jiang, H.; Qiao, Y.; et al. A Method for Obtaining 3D Point Cloud Data by Combining 2D Image Segmentation and Depth Information of Pigs. *Animals* **2023**, *13*, 2472. https://doi.org/10.3390/ani13152472.
17. Chang, K.; Ma, W.; Xu, X.; et al. A Point Cloud Segmentation Method for Pigs from Complex Point Cloud Environments Based on the Improved PointNet++. *Agriculture* **2024**, *14*, 720. https://doi.org/10.3390/agriculture14050720.
18. Zhao, W.; Zhang, R.; Wang, Q.; et al. BFANet: Revisiting 3D Semantic Segmentation with Boundary Feature Analysis. In Proceedings of the 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 11–15 June 2025; pp. 29395–29405. https://doi.org/10.1109/cvpr52734.2025.02737.
19. Tang, L.; Zhan, Y.; Chen, Z.; et al. Contrastive Boundary Learning for Point Cloud Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 8479–8489. https://doi.org/10.1109/cvpr52688.2022.00830.
20. Lu, T.; Wang, L.; Wu, G. CGA-Net: Category Guided Aggregation for Point Cloud Semantic Segmentation. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 11688–11697. https://doi.org/10.1109/cvpr46437.2021.01152.
21. Hu, H.; Yu, J.; Yin, L.; et al. An improved PointNet++ point cloud segmentation model applied to automatic measurement method of pig body size. *Comput. Electron. Agric.* **2023**, *205*, 107560. https://doi.org/10.1016/j.compag.2022.107560.
22. Peng, W.; Liu, Z.; Cai, J.; et al. Research and application progress of electronic ear tags as infrastructure for precision livestock industry: A review. *Intell. Robot.* **2025**, *5*, 433–449. https://doi.org/10.20517/ir.2025.22.
23. Pagin, G.; Martello, L.S.; Tabile, R.A.; et al. Proposing a framework for body mass prediction with point clouds: A study applied in typical swine pen environments. *Smart Agric. Technol.* **2025**, *12*, 101200. https://doi.org/10.1016/j.atech.2025.101200.
24. Zhang, Z. Flexible camera calibration by viewing a plane from unknown orientations. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; Volume 1, pp. 666–673. https://doi.org/10.1109/iccv.1999.791289.
25. Wang, P.S.; Liu, Y.; Guo, Y.X.; et al. O-CNN: Octree-based convolutional neural networks for 3D shape analysis. *ACM Trans. Graph.* **2017**, *36*, 1–11. https://doi.org/10.1145/3072959.3073608.
26. Zhou, K.; Gong, M.; Huang, X.; et al. Data-Parallel Octrees for Surface Reconstruction. *IEEE Trans. Vis. Comput. Graph.* **2011**, *17*, 669–681. https://doi.org/10.1109/tvcg.2010.75.
27. Wang, J.; Sun, K.; Cheng, T.; et al. Deep High-Resolution Representation Learning for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 3349–3364. https://doi.org/10.1109/tpami.2020.2983686.
28. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. https://doi.org/10.1109/cvpr.2015.7298965.
29. Milletari, F.; Navab, N.; Ahmadi, S.A. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 565–571. https://doi.org/10.1109/3dv.2016.79.
30. Peng, B.; Wu, X.; Jiang, L.; et al. OA-CNNs: Omni-Adaptive Sparse CNNs for 3D Semantic Segmentation. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024; pp. 21305–21315. https://doi.org/10.1109/cvpr52733.2024.02013.
31. Wijaya, K.T.; Paek, D.H.; Kong, S.H. Advanced Feature Learning on Point Clouds Using Multi-Resolution Features and Learnable Pooling. *Remote Sens.* **2024**, *16*, 1835. https://doi.org/10.3390/rs16111835.
32. Wu, X.; Jiang, L.; Wang, P.S.; et al. Point Transformer V3: Simpler, Faster, Stronger. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024; pp. 4840–4851. https://doi.org/10.1109/cvpr52733.2024.00463.