RdmkNet & Toronto-Rdmk: Large-Scale Datasets for Road Marking Classification and Segmentation

Jing Du, Lingfei Ma, *Member, IEEE*, Jing Li, Nannan Qin, John Zelek, *Member, IEEE*, Haiyan Guan, *Senior Member, IEEE*, and Jonathan Li, *Fellow, IEEE*

***Abstract*— Effective road marking classification and segmentation play a pivotal role in advancing vehicle-to-everything (V2X) applications and refining road inventory databases. However, the irregular data formats and unordered permutation modes of 3D point clouds, along with the limited availability of large-scale datasets with point-level annotations, remain significant obstacles to designing deep learning-based networks with superior performance. To address these challenges, this paper proposes a novel multi-level feature optimization network structure, named MFPNet, and introduces two point cloud benchmarks, RdmkNet and Toronto-Rdmk, for road marking classification and segmentation in intricate urban environments. MFPNet is composed of three integral modules. First, the M-transformer module, consisting of three transformers obtained from different channels, fully captures rich point cloud background information and long-distance dependencies between objects. Then, the feature pooling aggregation module uses parallel structured pooling attention mechanisms to aggregate features captured by the M-transformer module, while the prediction refinement module further enhances the acquisition of semantic features. Comparative studies indicate that MFPNet can be embedded into general deep learning networks without changing their original network structures, significantly improving the accuracy of multiple baseline networks. Furthermore, extensive experiments demonstrate that the two newly-developed point cloud datasets are meaningful for road marking classification and segmentation tasks, contributing to the development of autonomous driving.**

***Index Terms*— Urban point cloud dataset, road marking, semantic segmentation, classification, transformer.**

Manuscript received 3 May 2023; revised 1 November 2023 and 25 January 2024; accepted 10 April 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 42101451, Grant 41971414, and Grant 42001400; in part by China Scholarship Council under Ph.D. Scholarship under Grant 202208350003; in part by the Startup Foundation for Introducing Talent of Nanjing University of Information Science and Technology (NUIST) under Grant 2023r093; and in part by the Emerging Interdisciplinary Project of Central University of Finance and Economics in China. The Associate Editor for this article was Q. Zhang. *(Corresponding authors: Lingfei Ma; Haiyan Guan.)*

Jing Du and John Zelek are with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: j7du@uwaterloo.ca; jzelek@uwaterloo.ca).

Lingfei Ma is with the School of Statistics and Mathematics, Central University of Finance and Economics, Beijing 102206, China (e-mail: l53ma@cufe.edu.cn).

Jing Li is with the School of Information, Central University of Finance and Economics, Beijing 102206, China (e-mail: lijing2017@cufe.edu.cn).

Nannan Qin and Haiyan Guan are with the School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: nnqin@nuist.edu.cn; guanhy.nj@nuist.edu.cn).

Jonathan Li is with the Department of Geography and Environmental Management and the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: junli@uwaterloo.ca).

Digital Object Identifier 10.1109/TITS.2024.3394481

I. Introduction

3D POINT cloud classification and semantic segmentation are crucial and challenging tasks in the field of computer vision and remote sensing, with various applications, including autonomous driving, robotics, augmented reality, high-definition map construction, and urban mapping [1], [2]. The objective of semantic segmentation of point clouds is to assign a semantic label to each 3D point, while the goal of 3D point cloud classification is to assign a class label to each point cloud, with all points in the point cloud sharing one class label. Hence, classification and semantic segmentation can provide fine-grained scene understanding and complement the task of object detection [3]. Deep learning has proven to be an effective and efficient solution for point cloud classification and semantic segmentation tasks. However, training deep learning models for classification and semantic segmentation is demanding and resource-intensive, requiring large-scale datasets with point-level annotations.

Advances in sensor technology, particularly in the rapid development of diverse LiDAR systems (ground-based, airborne, satellite-based) and other devices such as contact scanners and depth cameras, have significantly enhanced the flexibility and convenience of acquiring point cloud data. Currently, most publicly available point cloud semantic segmentation datasets are scanned using radar sensors, including the SICK LMS, Riegl LMS-Q120i, Terrestrial Laser Scanner, Velodyne HDL-32E, Velodyne HDL-64E, Teledyne Optech Maverick, and others [4]. However, even with a variety of available data acquisition methods and decreasing difficulty of data acquisition, the process of labeling for intensive datasets is still very challenging and time-consuming [5], [6]. Although some annotation tools have been developed, there is still a lack of fully automated and intelligent annotation tools that can meet the data requirements for deep learning. Data annotation still relies heavily on manual work. Most point cloud segmentation and classification datasets contain at least 10 million points, rendering this labeling process extremely tedious. In summary, despite the growing availability of point cloud data, the process of labeling dense datasets remains a significant bottleneck in designing novel learning-based networks with superior performance.

Currently, the most popular publicly available semantic segmentation datasets include S3DIS [7], ScanNet [8], Semantic3D [9], Paris-Lille-3D [10], SemanticKITTI [11], Toronto-3D [4], and others. Almost all object classification and

semantic segmentation networks employed these datasets for algorithm development and performance comparison. However, these datasets do not specifically focus on road markings and road surroundings.

The S3DIS dataset [7] and ScanNet [8] dataset are indoor datasets. The Semantic3D dataset [9] contains eight semantic classes: man-made terrain, natural terrain, high vegetation, low vegetation, buildings, remaining hardscapes, scanning artifacts, and cars. It covers a wide range of urban outdoor artificial scenes and rural natural scenes, such as churches, streets, town halls, sports grounds, villages, football fields, and castles. The Semantic-KITTI dataset [11] shows inner-city traffic and residential areas around Karlsruhe, Germany, as well as highway scenes and rural roads. It includes 28 semantic categories, but there is no finer distinction between road markings, only sidewalk categories. It is worth mentioning that the entire dataset consists of 518 tiles, requiring over 1400 hours of annotation work. Additionally, each tile needs 10 to 60 minutes of validation and correction, resulting in a total of over 1700 hours. The Toronto-3D dataset [4] covers approximately 1 km of roads in Toronto, Canada, and comprises approximately 78.3 million points. The dataset was manually labeled into eight categories: roads, road markings, nature, buildings, utility lines, poles, cars, and fences. Currently, there is a limited availability of commonly used classification data, mainly the 3D CAD model dataset ModelNet40 [12]. It contains 40 common object categories, such as bed, bench, bookshelf, bottle, chair, sink, sofa, stair, stool, table, and more. The ModelNet40 dataset focuses on indoor scenes.

Therefore, it is highly noteworthy to create new dataset benchmarks based on existing ones that match specific application scenarios. In the realm of intelligent transportation systems, road markings, characterized by their distinct shapes and sizes, play an instrumental role in ensuring vehicular safety and traffic regulation. While numerous datasets exist for classification and semantic segmentation in autonomous driving, there remains a notable deficiency in datasets specifically tailored to road markings and their adjacent environments. To address this deficit, the Toronto-Rdmk dataset was meticulously constructed in this paper by integrating the Toronto-3D dataset [4], recognized for its expansive road surroundings data, with the 3D road marking dataset [13], which offers a vast collection of road marking details. Concurrently, the structure of the dataset was adapted to mirror real-world scenarios more closely. Categories were refined to be comprehensive, with broader categories encapsulating diverse road markings to enhance neural network training efficiency. In parallel, RdmkNet was developed to rectify the limitations inherent in the original road marking dataset. Recognizing the intricate relationships between individual road markings in real-world driving scenarios, the dataset was restructured to better capture and reflect these complexities. Categories were augmented and amalgamated to address the noticeable gaps in the foundational dataset.

Accurate classification and segmentation of road markings are imperative for the advancement of V2X applications. The emergence of autonomous driving further underscores the necessity of precise road marking extraction [14]. Traditional methods, which emphasized manually crafted features from 3D point clouds, though resourceful, often found their efficacy compromised in intricate urban settings [15]. In contrast, contemporary research has leaned heavily into deep learning paradigms for road marking segmentation. For instance, in addressing the complexities of multi-beam data, a study [16] advocated for a distinctive method that harnesses point clouds from budget-friendly mobile LiDAR setups. This methodology combines a pseudo-scan line structure with a density-balanced window median filter and a marker edge constraint detection technique, showcasing impressive road marking detection prowess. Another groundbreaking investigation [17] introduced a deep learning framework leveraging capsule networks for road marking extraction and classification from mobile LiDAR point clouds. Traditional techniques face challenges due to point density and intensity variations in LiDAR data. The proposed framework, comprising data-preprocessing, extraction, and classification modules, addresses these challenges, improving efficiency and robustness. Subsequent research [18], determined to outclass conventional strategies, introduced a model integrating a dense feature pyramid network (DFPN) and a focal loss function, setting new standards in marking extraction. Adding to this array of innovations, the GAT_SCNet approach [19] capitalized on the graph attention network's multi-head attention dynamic, achieving stellar results in discerning road markings' intricate spatial interrelationships. Acknowledging the pivotal role of road marking accuracy, another initiative [20] assembled an dataset of Spanish road markings and leveraged deep learning to detect marking damages. Lastly, a study [21] introduced the multi-attentional semantic segmentation (MASS) framework. This framework, optimized for detailed top-view interpretations, is based on a multi-attention mechanism and the PillarSegNet method. It contributes to redefining benchmarks in top-view LiDAR data segmentation.

While LiDAR offers a groundbreaking shift from traditional methods by capturing detailed and environment-resilient point cloud data, further optimization is necessary to truly harness its potential. Current methods, though advanced, still face challenges in establishing spatial relationships between data points, and in extracting and representing features in a manner that can be efficiently utilized for large-scale point cloud analysis. This leads us to the question: can we devise a more effective mechanism to interpret and utilize the rich data that LiDAR provides, especially in the context of road marking segmentation? Building upon the strengths of LiDAR and recognizing the gaps in existing methodologies, in this paper, we introduce a novel approach to address these challenges.

Therefore, we propose the M-transformer module to establish spatial relationships between points in a straightforward and effective manner to achieve feature extraction. Additionally, we propose the feature pooling aggregation module, which obtains the critical features in the global features through max-pooling and average-pooling operations. The weights of the features are learned by the softmax function and multiplied by the input features. The prediction refinement module obtains the maximum and average values of the input features and then concatenates them to obtain the results.

Similarly, the feature weights are acquired by the softmax function, and afterward, the feature weights are calculated with the initial features. The M-transformer module, the feature pooling aggregation module, and the prediction refinement module constitute the multi-level transformer feature optimization network architecture MFPNet. The proposed MFPNet can be easily embedded into other networks to enhance feature representation for large-scale point cloud analysis tasks.

Our contributions can be summarized as follows: (1) We introduce two new point cloud datasets, i.e., RdmkNet and Toronto-Rdmk, which are specifically designed for autonomous driving scenarios. These datasets are suitable for road marking classification and semantic segmentation tasks, respectively. (2) We propose MFPNet, a novel multi-level transformer feature optimization network structure that enhances feature representation for large-scale point cloud analysis tasks. This network structure can be easily integrated into other networks without changing their original network structures. (3) We perform extensive experiments on the Toronto-Rdmk, Toronto-3D, and RdmkNet datasets, respectively. These experiments demonstrate the superior performance of MFPNet and validate the practical training significance of the two proposed datasets.

II. Related Work

A. Object Classification

Deep learning methods for 3D object classification can be primarily categorized into multiview-related, voxel-related, and point-related approaches.

Multiview-related approaches [22], [23], [24], [25] first classify 3D objects by projecting the 3D shapes into several 2D views and then learn features from each view to achieve accurate shape classification through feature fusion. These methods can effectively capture rich texture information from the increasing number of views of 3D objects, contributing to high classification accuracy. However, multiview-related approaches tend to overlook the connections that exist between regions and views among a variety of view images. These relationships, however, are crucial for the representation of 3D objects in a multi-view setting. To address this limitation, a relational network [23] was presented to enhance the knowledge of separate view images by efficiently connecting the areas corresponding to various viewpoints, thus utilizing the interrelationships among a set of views and aggregating such views to achieve discriminative 3D representations.

Voxel-related approaches [26], [27], [28] classify 3D objects by converting a point cloud into a voxel grid representation, followed by utilizing 3D CNNs for object classification. Moreover, such methods are capable of handling 3D objects of different sizes and shapes, which could properly deal with the more general classification problem. Point-related approaches [29], [30], [31], [32] can directly consume 3D point clouds for 3D object classification without changing the data format, thus avoiding unnecessary information loss. Hassan et al. [32] proposed a module amalgamating sampling, pooling, and annular convolution layers to effectively aggregate point features. Enhanced by a residual block with skip connections and non-linear shortcuts, this method supported hierarchical learning directly from raw point clouds.

B. Semantic Segmentation

Deep learning methods for 3D semantic segmentation are normally divided into four categories: projection-related [33], [34], [35], [36], [37], discretization-related [38], [39], [40], [41], [42], point-related [43], [44], [45], [46], [47], and hybrid methods [48], [49], [50], [51].

Projection-related approaches convert raw point clouds into regular formats like multiple views for semantic segmentation. Such methods are inevitably influenced by viewpoint selection and occlusion, resulting in significant information loss, including internal geometry information. Discretization-related methods generally start by converting point clouds into discrete representations, such as 3D voxels, which are processed by 3D convolutional networks. Consequently, all points in the voxel are given the same semantic label as the voxel to acquire the point-level semantic segmentation results [38]. However, a regular voxel data representation has many empty voxels, leading to reduced computational efficiency when applying dense convolutions to such empty voxels. Graham et al. [40] addressed this issue by introducing a submanifold sparse convolution (SSC) operator, which limited the convolution outputs to occupied voxels, resulting in significant reductions in memory and computational costs. SSC is highly efficient in processing sparse data in high-dimensional spaces compared to conventional 3D convolution operations, which has led to its widespread adoption [52], [53].

Point-related networks operate directly on 3D point clouds, despite the fact that they are disorganized and unstructured. Hence, applying standard CNNs on point cloud data is not feasible. PointNet [43], as the landmark algorithm in the direction of semantic segmentation, achieved rotation invariance and permutation invariance of the disordered point clouds through spatial transformation and max-pooling operations. Since then, numerous point-related networks have been introduced, and these have been broadly categorized into point-by-point MLP methods, point convolution techniques, recurrent neural networks (RNNs)-related methods, and graph-related methods. For example, the SCF-Net [54] addressed the challenge of capturing spatial contextual information in an unstructured data environment. SCF-Net introduced the SCF module with three blocks: Local Polar Representation (LPR), Dual-Distance Attentive Pooling (DDAP), and Global Contextual Feature (GCF). These components respectively tackle local context representation, local feature learning, and global contextual feature capture, overcoming issues like z-axis rotation variance in local contexts. Hybrid methods combine the strengths of the aforementioned methods and attempt to overcome their shortcomings. RNNs are employed for semantic segmentation as well to extract intrinsic contextual features from point clouds. For instance, 3DCNN-DQN-RNN [48] was proposed, where residual RNN extracted and fused the 3DCNN features, coordinates, and colors of each point in multiple scales, resulting in a more robust and differentiated feature representation.

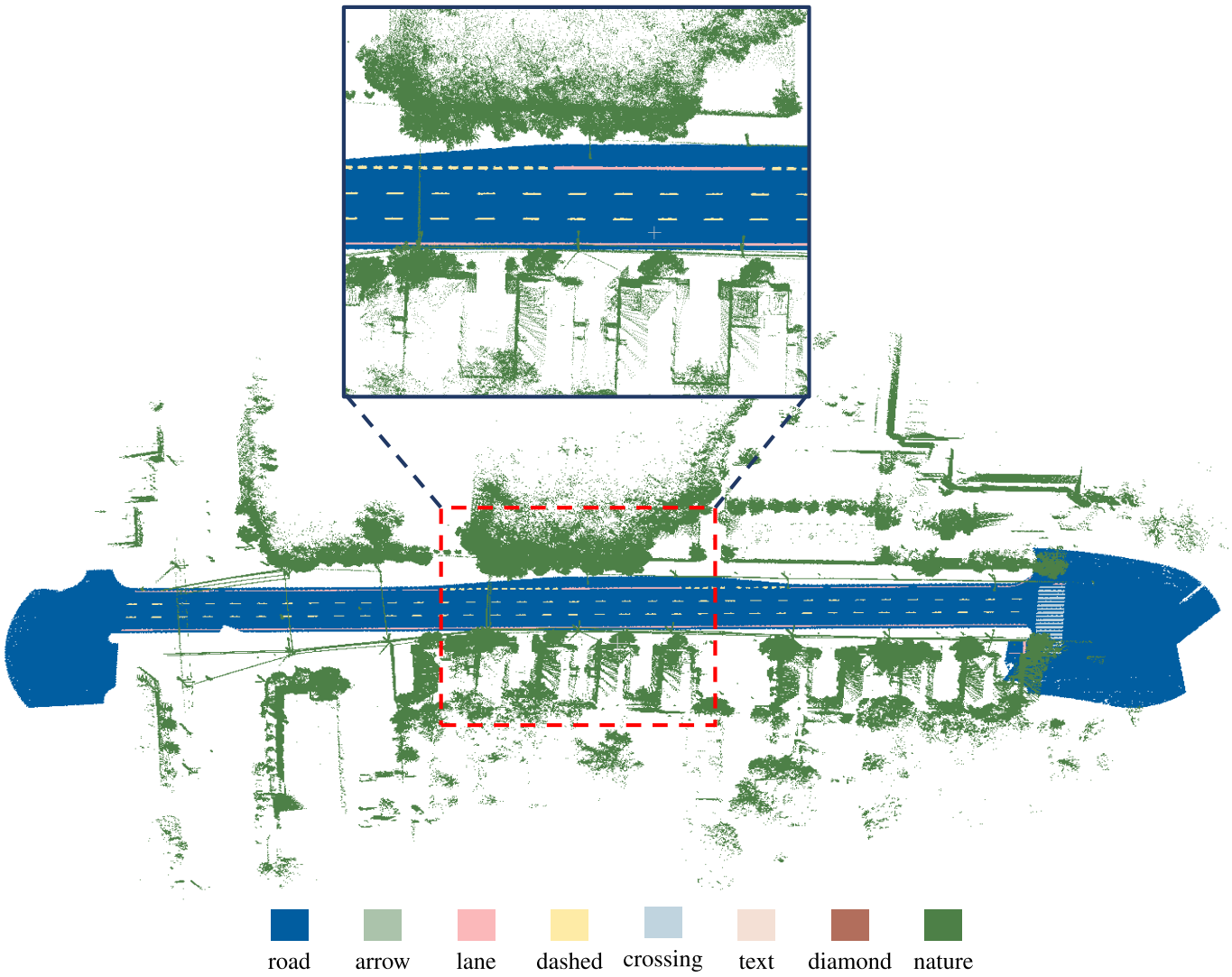

Fig. 1. Illustration of the TR_1 training set in the proposed Toronto-Rdmk dataset. Different colors signify distinct semantic categories in the dataset. The segment labeled as 'road' pertains to the primary drivable surface for vehicles. 'nature' encompasses a diverse range of environmental features including trees, vertical barriers such as wooden fences and construction site walls, parts of low and multi-story buildings, storefronts, and shrubs. The categories of 'arrow', 'lane', 'dashed', 'crossing', 'text', and 'diamond' each refer to specific road markings and indicators that convey critical navigational and regulatory information for drivers and autonomous systems alike.

C. Transformer

The Transformer architecture [55] initially gained widespread utilization in the domains of NLP [56], [57], [58] and image processing [59], [60], [61]. However, Transformer has now flourished in various domains, including the 3D computer vision field. The considerable challenge in processing point clouds is due to their disordered properties, but Transformer does not require a secondary operation to achieve permutation invariance.

Fan et al. [62] introduced the Single-stride Sparse Transformer (SST) to evaluate the influence of different stride sizes on 3D object detectors, leveraging local attention to mitigate the challenge of a shrinking receptive field. Lai et al. [63] implemented a hierarchical sampling strategy in transformer-based networks, densely sampling close points and sparsely sampling distant ones, enhancing the receptive field and contextual information with reduced computational costs. Zhao et al. [64] devised GraFormer, a transformer and graph convolution-based approach for 3D pose estimation. It used GraAttention for global interaction among 2D joints and ChebGConv to highlight implicit relationships between joints. Lastly, Wang et al. [65] presented several modules including, Geometric Details Perception (GDP) and Self-Feature Augmentation (SFA), to perceive short and long-term relationships among points. A new framework for point cloud completion using a popular encoder module is built on the foundation of GDP and SFA, which addresses the impact of data density distribution and produces high-quality complete shapes.

III. New Datasets

A. Toronto-Rdmk

Toronto-Rdmk consists of the Toronto-3D dataset [4] and the 3D road marking dataset [13]. The Toronto-3D dataset contains rich road surroundings points, and the 3D road markings dataset includes abundant road marking points. More specifically, we extracted five types of 3D points from the

TABLE I
THE NUMBER OF POINTS FOR DIFFERENT CLASSES IN EACH SCENE

Scene	Road	Arrow	Lane	Dashed	Crossing	Text	Diamond	Nature	Total
TR_0	11,064,929	36,707	248,690	116,055	36,089	627	0	1,944,890	13,447,987
TR_1	12,068,636	0	212,177	90,126	129,946	0	0	1,944,890	14,445,775
TR_2	10,210,885	0	313,874	107,879	0	0	0	14,650,765	25,283,403
TR_3	20,554,980	0	615,532	80,770	686,886	0	0	3,062,912	25,001,080
TR_4	13,758,316	0	385,500	98,803	335,988	13,232	0	3,062,912	17,654,751
TR_5	18,164,480	37,943	810,200	65,646	645,926	0	29,174	3,062,912	22,816,281
TR_6	8,539,945	0	97,638	183,245	0	1,006	0	7,997,941	16,819,775
TR_7	12,282,113	0	4,414	27,281	0	52	13,534	7,997,941	20,325,335
TR_8	8,222,745	0	199,003	95,304	191,945	0	11,398	7,997,941	16,718,336
TR_9	6,919,193	0	146,793	117,808	0	0	0	14,650,765	21,834,559
TR_10	4,701,327	0	53,685	78,915	0	0	0	14,650,765	19,484,692
TR_11	5,754,153	87,711	230,590	46,133	0	0	0	3,062,912	9,181,499
Total	132,241,702	162,361	3,318,096	1,107,965	2,026,780	14,917	54,106	84,087,546	223,013,473

Toronto-3D dataset, i.e., nature, buildings, power lines, poles, and fences. As we primarily focus on the applications for autonomous driving road surface scenarios, road surroundings are necessary but relatively less important. Thus, we combined the data from the above-mentioned five categories into the nature class. Meanwhile, we extracted the road surface data from the 3D road marking dataset, which contains a rich collection of road markings, including arrows, lanes, dashed lines, zebra crossings, text, and diamond-shaped signs. The original 3D road markings dataset contains arrows, dashed lines, and texts, which are further classified into finer categories. The length of dashed lines 1 and 2 are dissimilar, while texts 1, 2, 3, and 4 correspond to distinct Chinese characters. Moreover, arrows 1 and 2 denote straight and turning arrows, respectively. However, the limited number of instances in each of these categories is not sufficient for effective network training. Thus, we merged the arrows, dashed lines, and texts into broader categories. Specifically, we combined dashed lines 1 and 2 into a single dashed lines category, texts 1, 2, 3, and 4 into a unified texts category, and road arrows 1 and 2 into one road arrows category. Consequently, Toronto-Rdmk contains eight classes: road, arrow, lane line, dashed line, zebra crossing, text, diamond, and nature.

The 3D road marking dataset and the Toronto-3D dataset contain twelve road scenes and four road scenes, respectively. To adhere to the distribution rules of the Toronto-3D dataset, we included L001, L003, and L004 in the training set, while L002 was assigned to the testing set. For the 3D road marking dataset, we selected eight scenes for the training set and four scenes for the testing set. Next, we combined the training sets of both datasets to create a unified training set. Similarly, we merged the testing sets from the two datasets to create a testing set. Fig. 1 shows the training set TR_1 scene with a zoom-in view, while Table I presents the number of points for the different classes in each scenario.

B. RdmkNet

For the task of road marking classification, the original dataset [13] manifests several critical limitations. As delineated in Fig. 2, the dataset segregates the dashed lines, lane lines, and zebra crossings into separate files, with each file predominantly showcasing a single type of line. However, the true differentiation among categories such as dashed lines, lane lines, and zebra crossings arises from the unique arrangements of these lines rather than the individual lines themselves. Further analyzing the original 3D road markings dataset, we observed it encompassed arrows, dashed lines, and texts. These were further refined into subcategories: two distinct

Fig. 2. Illustration of the three main representative categories from the original road marking dataset: zebra crossing, dashed line, and lane line. Each type is isolated in distinct files, focusing on individual line entities. This presentation limits the representation of their inherent interplay and configurations, crucial for real-world autonomous driving road marking classification.

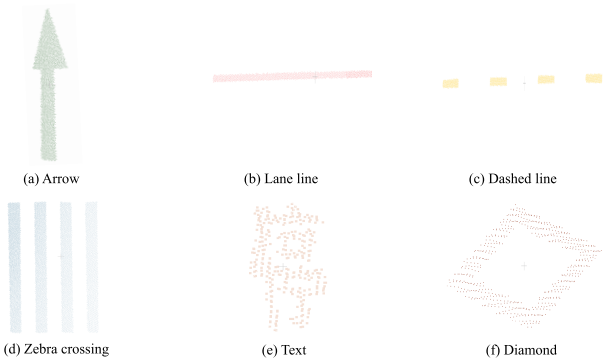

Fig. 3. Illustration of the six refined categories in the RdmkNet dataset: arrow, lane line, dashed line, zebra crossing, text, and diamond. The dataset underwent a rigorous refinement process, including re-extraction of road markings and restructuring to suit each category's unique shape. Instead of maintaining numerous subcategories, we merged them to form these broader, more discernible categories. Due to the limited instances of the triangle category, it was excluded. Additionally, to bolster the dataset, we integrated rich data elements from the 3D road marking dataset. This enhancement aims to achieve a higher fidelity representation of real-world road scenarios, with a focus on ensuring data quality and precision for advanced classification tasks.

TABLE II
THE NUMBER OF OBJECTS FOR DIFFERENT CLASSES IN THE TRAINING SET AND TEST SET OF THE PROPOSED RDMKNET DATASET

	Arrow	Crossing	Dashed	Diamond	Lane	Text	Total
Training set	525	1,470	3,604	489	3,814	385	10,287
Test set	225	630	1,546	211	1,636	165	4,413
Total	750	2,100	5,150	700	5,450	550	14,700

lengths for dashed lines, varied Chinese characters for texts, and specific designations for arrows (e.g., straight and turning). However, the limited number of instances in these categories hindered effective network training. Accordingly, we merged the arrows, dashed lines, and texts into broader categories, combining similar entities into singular categories for arrows, dashed lines, and texts. Additionally, the original dataset showed a glaring absence of triangle-shaped markings. Even if data augmentation techniques can be employed to enrich this collection, they may still fall short of the voluminous data prerequisites of deep learning models, limiting for training robust deep neural networks.

To rectify these challenges, we implemented comprehensive data processing measures. We re-extracted road markings from every scene, allowing us to reorganize and tailor the data to each category's specific shape. Recognizing the limitations of finer subcategories in the original dataset, we merged the arrows, dashed lines, and texts into broader categories, combining similar entities into singular categories. This heightened the dataset's quality, making it more fit for precise classification tasks. Post-restructuring, we classified the data into six clear-cut categories: arrow, lane line, dashed line, zebra crossing, text, and diamond, as depicted in Fig. 3. Addressing the dataset's noticeable scarcity in the triangle domains, we opted to eliminate the triangle category. Moreover, we harnessed data from the 3D road marking dataset, which boasted an abundant array of road markings, like arrows, lanes, and diamond-shaped signs. To mitigate the scarcity of road marking instances in the original collection, we employed data augmentation, choosing 50 random rotation orientations, thereby augmenting our dataset fiftyfold. In this paper, we divided the enhanced dataset into a 7:3 training-testing ratio, yielding 10,287 training and 4,413 testing samples, as detailed in Table II. Prioritizing data diversity, we enriched the dataset with normals and random noise, ensuring it mirrors the varied and unpredictable facets of real-world driving scenarios.

IV. METHOD

A. Motivation

(1) Critical Need for Precise Road Marking Interpretation. Accurate segmentation of road markings plays an instrumental role in enhancing V2X applications and regularly updating road inventory databases. The precise delineation of these markings directly contributes to the safety of autonomous driving systems within diverse road scenarios. Nonetheless, the inherent irregularity of 3D point cloud data, combined with unordered permutation patterns and a paucity of point-level annotated datasets, poses substantial challenges in the development of effective deep learning-based networks.

(2) Leveraging Advanced Neural Structures in Point Cloud Processing. The advent and prominence of Transformer-based models in diverse computational domains suggest their potential utility in enhancing point cloud data processing. Effectively amalgamating these transformative architectures into existing deep learning networks, without perturbing their foundational structures, remains an area worthy of exploration.

(3) Enriching Representations with Low Attention Scores. Current methodologies, especially Transformer-based approaches, are predisposed to marginalizing point relationships with lower attention scores. However, even low-scoring relationships are crucial for understanding certain road scenarios. The challenge lies in developing a mechanism that factors in these overlooked relationships, thereby refining the prediction and leading to a more comprehensive understanding of the environment.

Algorithm 1 MFPNet Processing Pipeline

Require: Input point cloud: $\mathcal{P} \in R^{N\times d}$ with features $f_i \in R^d$
Ensure: Refined features: $\mathcal{F}_{Ref}$

function M-TRANSFORMER($\mathcal{P}$)
  $\mathcal{F}_0 \leftarrow$ InitialFeatures($\mathcal{P}$)
  $\mathcal{F}_h \leftarrow \mathcal{T}_e(\mathcal{F}_0)$
  $\mathcal{M}_{Q,K,V} \leftarrow \mathcal{T}_l(\mathcal{F}_h)$
  Obtain multi-level $M_{Q,K,V}$ using Eq. (3)
  $\mathcal{F}^e_{(t)} \leftarrow \Phi_{dot}\left(Q_{(t)}, K_{(t)}\right)$
  Calculate $\mathcal{F}^{(t)}_A$ and $\mathcal{F}^m_A$ using Eqs. (8, 9)
  $\mathcal{F}_M \leftarrow$ Concat $\left(\mathcal{F}^1_A, \mathcal{F}^2_A\right)$
  return $\mathcal{F}_M$
function FEATUREPOOLINGAGGREGATION($\mathcal{F}_M$)
  Apply max-pooling and average-pooling on $\mathcal{F}_M$ to obtain $\mathcal{F}_{MA}$ using Eqs. (11, 12, 13)
  $\mathcal{F}_{Agg} \leftarrow Softmax(\mathcal{F}_{MA}) \cdot \mathcal{F}_M$
  return $\mathcal{F}_{Agg}$
function PREDICTIONREFINEMENT($\mathcal{F}_{Agg}$)
  Calculate $\mathcal{F}_G$ using Eq. (15)
  $\mathcal{F}_{Ref} \leftarrow Softmax(\mathcal{F}_G) \cdot \mathcal{F}_{Agg}$
  return $\mathcal{F}_{Ref}$
$\mathcal{F}_M \leftarrow$ M-TRANSFORMER($\mathcal{P}$)
$\mathcal{F}_{Agg} \leftarrow$ FEATUREPOOLINGAGGREGATION($\mathcal{F}_M$)
$\mathcal{F}_{Ref} \leftarrow$ PREDICTIONREFINEMENT($\mathcal{F}_{Agg}$)

B. Method Overview

The MFPNet comprises three essential modules: the M-Transformer module, the feature pooling aggregation module, and the prediction refinement module, as illustrated in

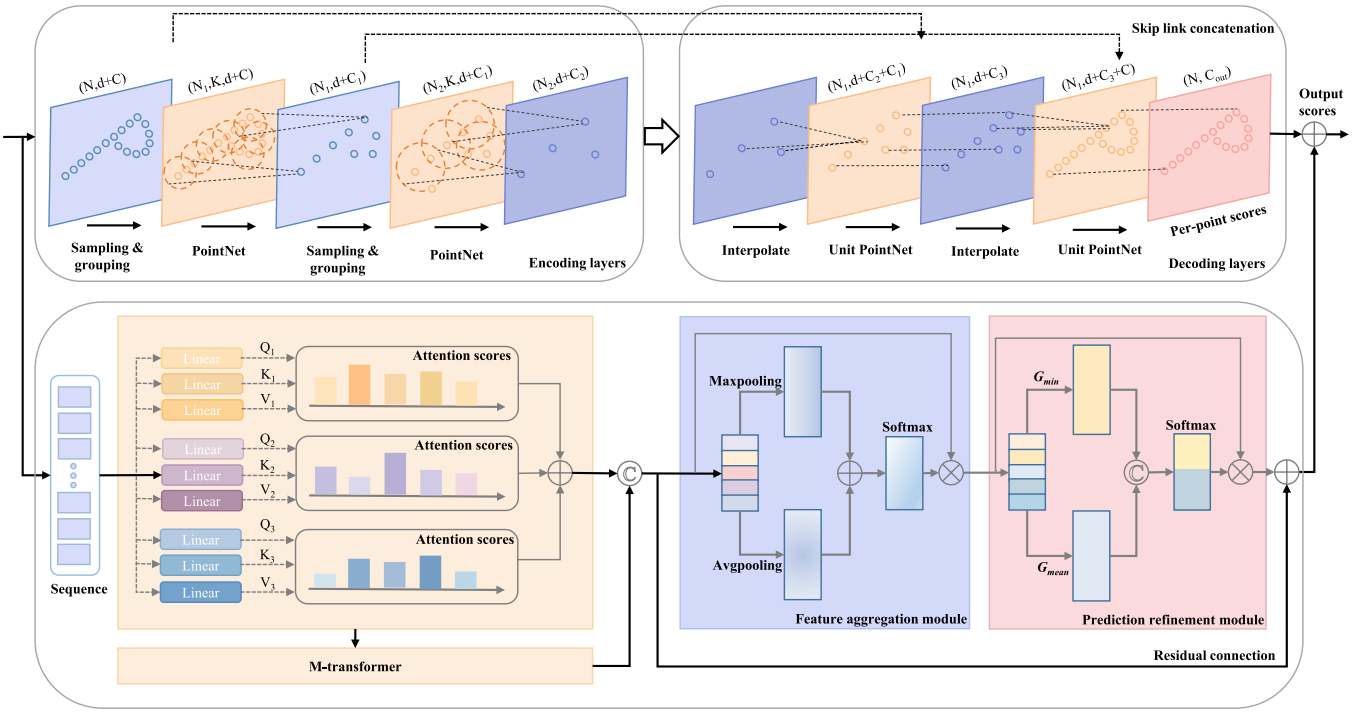

Fig. 4. Illustration of the proposed MFPNet (bottom) working on PointNet++ network (top). Within the PointNet++ structure, N signifies the number of input points, d stands for the point coordinates' dimensionality, and C denotes the dimensionality of the point features. The matrix $N \times (d+C)$ encompasses inputs from N points, integrating the d-dimensional coordinates and C-dimensional point features. The term K designates the number of neighboring points of centroid points, while N_1 and N_2 represent the number of subsampled points post-processing. The matrix $N_1 \times (d + C_1)$ outputs the N_1 subsampled points combined with d-dimensional coordinates and the revised C_1-dimensional feature vectors, and similar logic applies for subsequent layers. Transitioning to the MFPNet module, Q is the query matrix dictating attention focus, K is the key matrix paired with Q to determine focus areas, and V is the value matrix containing the data processed by the attention mechanism. The symbols $\bigoplus$, $\odot$ and $\otimes$ represent addition, concatenation, and multiplication operations, respectively.

Fig. 4. These modules work together to provide a comprehensive understanding of point cloud data by encoding both local and global dependencies. Additionally, the prediction refinement module further enhances predictions by incorporating point relationships with lower attention scores. This holistic approach ensures that MFPNet captures an accurate representation of the data, leading to improved performance. Algorithm 1 outlines the primary steps and computations required to carry out the MFPNet's intricate processes, commencing from the initial input of the point cloud up to the final refined features.

C. M-Transformer

The M-Transformer module is introduced to enrich feature encodings by capturing rich contextual information and long-range dependencies of point clouds. This module applies multiple linear transformations to input point cloud features, resulting in a hierarchy of point cloud features, including three stratified query matrices, key matrices, and value matrices. The attention mechanism in this module learns the importance of different point relationships, enabling the model to focus on the most relevant points. Furthermore, the multi-level transformer structure balances accuracy and computational cost, making it an efficient and effective solution for feature encoding in point cloud analysis.

To be more specific, each point p_i has d-dimensional features $f_i \in R^d$ in the input point cloud $\mathcal{P} = \{p_i \,|i = 1, 2, \ldots, N\} \in R^{N\times d}$, where N represents the number of points in the input point cloud and d denotes the feature dimension of each point. The initial features of the complete point cloud can be represented as $\mathcal{F}_0 \in R^{N\times d}$. High-dimensional point cloud features $\mathcal{F}_h$ are obtained by point feature encoding $\mathcal{T}_e$ as follows:

$$\mathcal{F}_h = \mathcal{T}_e\left(\mathcal{F}_0\right), \mathcal{F}_h \in R^{N\times C} \tag{1}$$

where C represents the dimensionality of the features $\mathcal{F}_h$.

Next, the linear transformation $\mathcal{T}_l$ is applied to the features $\mathcal{F}_h$ to produce the query matrix Q, the key matrix K and the value matrix V as follows:

$$\mathcal{M}_{Q,K,V} = \mathcal{T}_l\left(\mathcal{F}_h\right) \tag{2}$$

To extract a rich hierarchy of point cloud features, the M-transformer module adopts several linear transformations of different feature channels acting on the features, resulting in three stratified query matrices, key matrices, and value matrices. These matrices are expressed using the following equations:

$$M_{Q,K,V} = \begin{cases} Q_1 \in R^{N\times\frac{C}{4}} & K_1 \in R^{N\times\frac{C}{4}} & V_1 \in R^{N\times\frac{C}{4}} \\ Q_2 \in R^{N\times\frac{C}{8}} & K_2 \in R^{N\times\frac{C}{8}} & V_2 \in R^{N\times\frac{C}{8}} \\ Q_3 \in R^{N\times\frac{C}{16}} & K_3 \in R^{N\times\frac{C}{16}} & V_3 \in R^{N\times\frac{C}{16}} \end{cases} \tag{3}$$

Additionally, the attention module is capable of learning the query vector, the key vector, and the value vector. Then, the energy function $\mathcal{F}^e_{(t)}$ is obtained by performing the dot

product operation Φ_{dot} on the query vector and the key vector as follows:

$$\mathcal{F}^{e}_{(t)} = \Phi_{dot}\left(Q_{(t)}, K_{(t)}\right), \mathcal{F}^{e}_{(t)} \in R^{N\times N} \tag{4}$$

Next, the attention scores $\mathcal{S}_{(t)}$ are calculated by normalizing the energy function to a finite range using the Softmax function, followed by the weights of the value vectors as follows:

$$\mathcal{S}_{(t)} = \mathcal{S}oftmax\left(\mathcal{F}^{e}_{(t)}\right) \tag{5}$$

$$W^{v}_{(t)} = \frac{S_{(t)}}{\sigma + S_{(t)}} \tag{6}$$

where σ indicates the bias.

Moreover, the encoded point features $\mathcal{F}^{(t)}_{w} \in R^{N\times C}$ are obtained by multiplying the value vectors with their weights as follows:

$$\mathcal{F}^{(t)}_{w} = W^{v}_{(t)} \cdot V_{(t)} \tag{7}$$

These point features are sequentially operated by convolution, batch normalization, and the ReLU activation function to acquire the attention features $\mathcal{F}^{(t)}_{A} \in R^{N\times C}$. The output features $\mathcal{F}^{m}_{A} \in R^{N\times C}$ of three attention layers are fused by the following equations:

$$\mathcal{F}^{(t)}_{A} = Relu\left(BN\left(Conv\left(\mathcal{F}^{(t)}_{w}\right)\right)\right) \tag{8}$$

$$\mathcal{F}^{m}_{A} = \mathcal{S}um\left(\mathcal{F}^{(1)}_{A}, \mathcal{F}^{(2)}_{A}, \mathcal{F}^{(3)}_{A}\right) \tag{9}$$

The depth of the M-Transformer, denoted by the two-layer structure, can be adjusted based on computational capacity to strike a balance between accuracy and computational efficiency. Specifically, the outputs from each M-Transformer, represented as $\mathcal{F}_M \in R^{N\times 2C}$, are concatenated in the subsequent manner as follows:

$$\mathcal{F}_M = Concat\left(\mathcal{F}^{1}_{A}, \mathcal{F}^{2}_{A}\right) \tag{10}$$

In this study, the proposed M-Transformer module can effectively obtain long-range dependencies between different points, significantly improving the feature encoding in complex road scenarios.

D. Feature Pooling Aggregation Module

The feature pooling aggregation module is then developed to enhance the M-transformer by encoding local and global dependencies between objects based on pooling operations. Specifically, it utilizes max-pooling and average-pooling operations on the M-transformer features to capture both the most salient features and the global characteristics of objects. This allows for a more comprehensive understanding of the point cloud data and leads to a more robust representation of objects.

To be more specific, the feature pooling aggregation module takes the obtained features $\mathcal{F}_M$ as inputs and performs two pooling operations, i.e., max-pooling and average-pooling, to aggregate the learned features. The max-pooling operation is designed to extract the salient features of the object, while the average-pooling operation focuses on capturing the global features. Then, the features $\mathcal{F}_{MA}$ obtained from both pooling operations are concatenated to form the final feature representation as follows:

$$\mathcal{F}_{mp} = Maxpooling\left(\mathcal{F}_M\right) \tag{11}$$

$$\mathcal{F}_{ap} = Avgpooling\left(\mathcal{F}_M\right) \tag{12}$$

$$\mathcal{F}_{MA} = \mathcal{S}um\left(\mathcal{F}_{mp}, \mathcal{F}_{ap}\right) \tag{13}$$

Next, the feature weights are obtained by applying the softmax function to the features $\mathcal{F}_{MA}$. These weights are used to compute the weighted sum of features $\mathcal{F}_M$, which results in the aggregated features $\mathcal{F}_{Agg}$. This is calculated using the following equation:

$$\mathcal{F}_{Agg} = Softmax\left(\mathcal{F}_{MA}\right) \cdot \mathcal{F}_M \tag{14}$$

Therefore, the proposed M-Transformer module can effectively obtain long-range dependencies between different points, while the feature pooling aggregation module makes the network consider the local nature of objects and local interactions, significantly improving the network's ability to encode both local and global dependencies between objects in complex road scenarios.

E. Prediction Refinement Module

To overcome the limitation of the M-Transformer module, which tends to overlook point relationships with low scores, the prediction refinement module is introduced. This module computes both the minimum and mean values of the feature sequence, concatenating them to produce two-dimensional features. By applying feature weighting to the initial features, the module creates enriched feature representations that incorporate information with lower attention scores. The refined features, denoted as $\mathcal{F}_{Agg}$, are processed through two separate functional functions. The first function calculates the minimum value of the feature sequence, while the second function calculates the mean value. The resulting minimum and mean values are then concatenated to capture low-scoring point relationships, which can be computed using the following equation:

$$\mathcal{F}_G = Concat\left(G_{min}\left\{\mathcal{F}_{Agg}\right\}, G_{mean}\left\{\mathcal{F}_{Agg}\right\}\right) \tag{15}$$

Next, the feature weights of $\mathcal{F}_G$ are obtained using the softmax function, and these weights are then applied to the initial features using feature weighting to obtain the final predicted features $\mathcal{F}_{Ref}$ as follows:

$$\mathcal{F}_{Ref} = Softmax\left(\mathcal{F}_G\right) \cdot \mathcal{F}_{Agg} \tag{16}$$

This process effectively enriches the feature representations, even for lower attention scores, thereby contributing to the feature encoding ability of the proposed MFPNet.

V. Experiments

A. Semantic Segmentation

1) Dataset Description: We performed extensive experiments on 3D scene segmentation leveraging two pivotal datasets: the Toronto-Rdmk dataset and the Toronto-3D dataset. The Toronto-Rdmk dataset encompasses 12 distinct road scenes with an aggregate of approximately 223 million

TABLE III
SEGMENTATION RESULTS (%) OF DIFFERENT METHODS ON TORONTO-RDMK DATASET

Method	OA	mA	mIoU	Road	Arrow	Lane	Dashed	Crossing	Text	Diamond	Nature
PointNet++ [44]	97.4	64.6	56.1	97.0	17.8	96.3	65.7	82.2	0.0	0.0	89.7
PointNet++ + MFPNet	99.5	81.8	80.8	99.9	68.2	99.3	99.9	82.1	0.0	100.0	97.2
PointNet++ MSG [44]	99.6	82.7	79.6	99.6	66.2	94.8	94.3	97.1	0.0	86.2	99.2
PointNet++ MSG + MFPNet	99.8	83.8	83.4	100.0	70.7	99.4	100.0	100.0	0.0	98.4	99.4
SPG [66]	96.6	52.6	49.9	95.9	0.0	24.3	86.2	94.6	0.0	0.0	98.0
SPG + MFPNet	96.8	53.0	51.2	96.1	0.0	25.8	92.4	97.1	0.0	0.0	98.2
KPConv [45]	-	-	48.6	99.8	0.0	97.5	31.3	60.3	0.0	0.0	99.7
KPConv + MFPNet	-	-	53.8	99.6	0.0	96.7	50.9	83.4	0.0	0.0	99.6
FPConv [67]	98.1	65.1	63.3	99.9	0.0	98.6	93.1	29.0	0.0	96.0	89.9
FPConv + MFPNet	99.8	78.5	78.2	99.9	28.8	98.8	100.0	98.7	0.0	100.0	99.2
PAConv [68]	99.7	62.4	61.9	99.9	0.0	99.3	98.2	99.5	0.0	0.0	98.3
PAConv + MFPNet	97.8	65.6	64.6	97.5	45.6	80.9	87.0	75.4	0.0	39.4	91.2
Point Transformer V2 [69]	94.5	69.2	68.3	79.7	0.0	91.5	94.7	64.2	96.8	26.6	93.0
Point Transformer V2 + MFPNet	98.7	71.1	70.6	99.4	0.0	96.6	100.0	80.1	0.0	92.1	96.8
GAM [70]	99.5	86.0	84.2	99.6	97.8	90.0	98.1	98.0	0.0	91.9	98.4
GAM + MFPNet	99.8	89.1	89.0	99.9	13.2	100.0	100.0	100.0	100.0	99.9	98.9

points. It differentiates data into eight salient categories: road, arrow, lane line, dashed line, zebra crossing, text, diamond, and nature. The Toronto-3D dataset, on the other hand, contains 1km of road scenes with around 78.3 million points and includes eight categories: road, road marking, nature, building, utility line, pole, car, and fence. Through rigorous experimentation on both datasets, we have substantiated the superior performance of the proposed MFPNet in semantic segmentation on vast road scenarios.

2) Experimental Setup: In setting up experiments, we largely adhered to the original parameter settings for each network. Nevertheless, due to GPU memory constraints, we made necessary adjustments in certain areas. Specifically, we reduced the batch size to manage the limited memory capacity effectively. Additionally, to further accommodate this constraint, we increased the size of the subsampling grid, which helped in reducing the computational load. Concurrently, a reduction in the maximum number of points processed in each batch was implemented. These modifications were carefully considered to ensure that the experimental setup remained viable within the limits of our hardware resources while striving to minimize any impact on the performance and accuracy of the networks. To maintain fairness in evaluations, we adopted consistent parameters for the same comparison networks. It is pertinent to highlight that further parameter refinements have the potential for better outcomes across various models.

3) Segmentation Results on the Toronto-Rdmk: Table III presents the experimental results of our proposed method, which we tested against multiple baseline models to compare their segmentation accuracy. We used the Toronto-Rdmk dataset and evaluated the results using three metrics, i.e., Overall Accuracy (OA), mean Accuracy (mA) and mean Intersection over Union (mIoU). Our method outperformed the multiple baseline networks, as demonstrated in Table III. Compared to the original PointNet++, the version of PointNet++ integrated with MFPNet demonstrated marked enhancements in OA and mA, with gains of 2.1% and 17.2% respectively. Notably, with MFPNet, the mIoU of PointNet++ experienced a significant increase, showing a growth of 24.7% to reach 80.8%. Additionally, PointNet++ + MFPNet achieved a 100% IoU in the diamond category. When compared to PointNet++ + MSG, PointNet++ + MSG + MFPNet showed improvements in all three evaluation metrics OA, mA, and mIoU by 0.2%, 1.1%, and 3.8%, respectively.

When integrated with MFPNet, SPG achieved enhancements of 0.2% in OA, 0.4% in mA, and 1.3% in mIoU compared to the standalone SPG. Similarly, KPConv augmented with MFPNet exhibited a substantial mIoU increase of 5.2% over the base KPConv. Furthermore, FPConv, when combined with MFPNet, marked improvements of 1.7% in OA, 13.4% in mA, and 14.9% in mIoU over its standalone counterpart. Notably, FPConv + MFPNet achieved 100% IoU in the segmentation of dashed lines and diamonds, outperforming the base FPConv. Likewise, PAConv integrated with MFPNet realized improvements of 3.2% and 2.7% in mA and mIoU metrics, respectively, when contrasted with the base PAConv. When integrated with the proposed MFPNet, Point Transformer V2 not only maintains its original strengths but also demonstrates marked improvements in segmentation performance. This indicates the effectiveness of MFPNet in complementing and amplifying the capabilities of advanced transformer-based models like Point Transformer V2, especially in complex urban road scenarios. GAM, illustrating the application of the Gradient Attention Module to the PointNet++ network, displayed notable advancements. The further integration of MFPNet into this configuration resulted in additional accuracy enhancements, with increases of 0.3% in OA, 3.1% in mA, and 4.8% in mIoU. Moreover, this combination led to notable improvements in the IoU across all categories. The proposed MFPNet leverages a hierarchical structure that helps in capturing both global and local

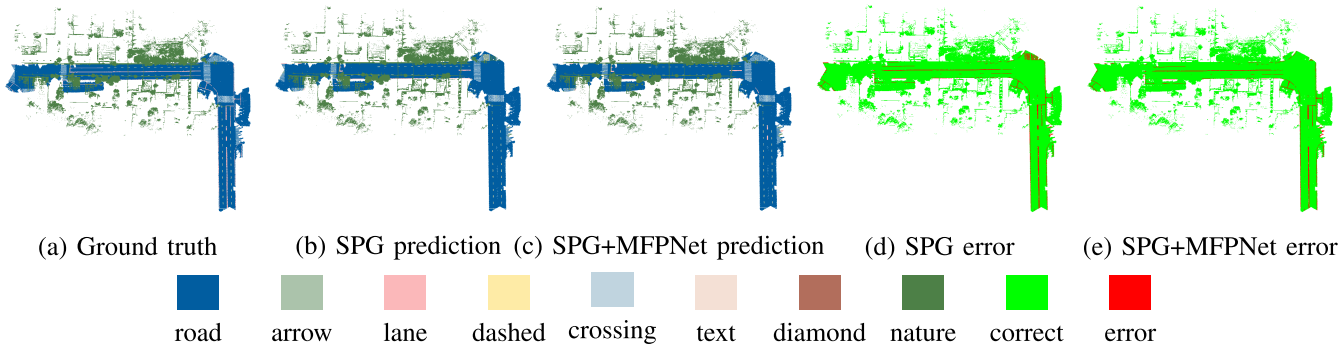

Fig. 5. Segmentation results on Toronto-Rdmk dataset. (a) Ground truth, (b) SPG, (c) SPG + MFPNet prediction, (d) SPG error, (e) SPG + MFPNet error. Diverse colors represent distinct semantic categories. Areas segmented accurately are denoted in green, while mis-segmented regions are marked in red.

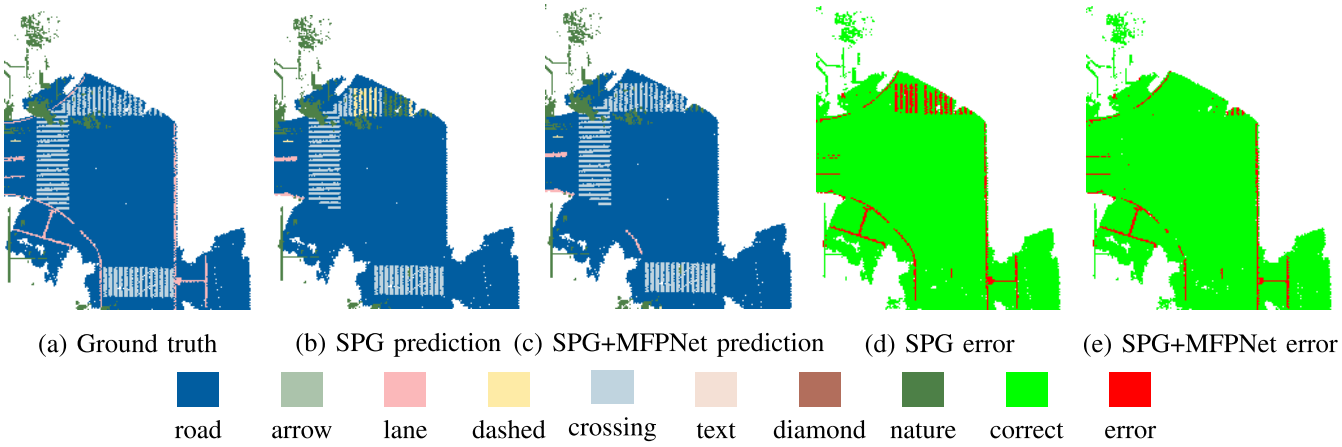

Fig. 6. Detailed segmentation results on Toronto-Rdmk dataset. (a) Ground truth, (b) SPG, (c) SPG + MFPNet prediction, (d) SPG error, (e) SPG + MFPNet error. Diverse colors represent distinct semantic categories. Areas segmented accurately are denoted in green, while mis-segmented regions are marked in red.

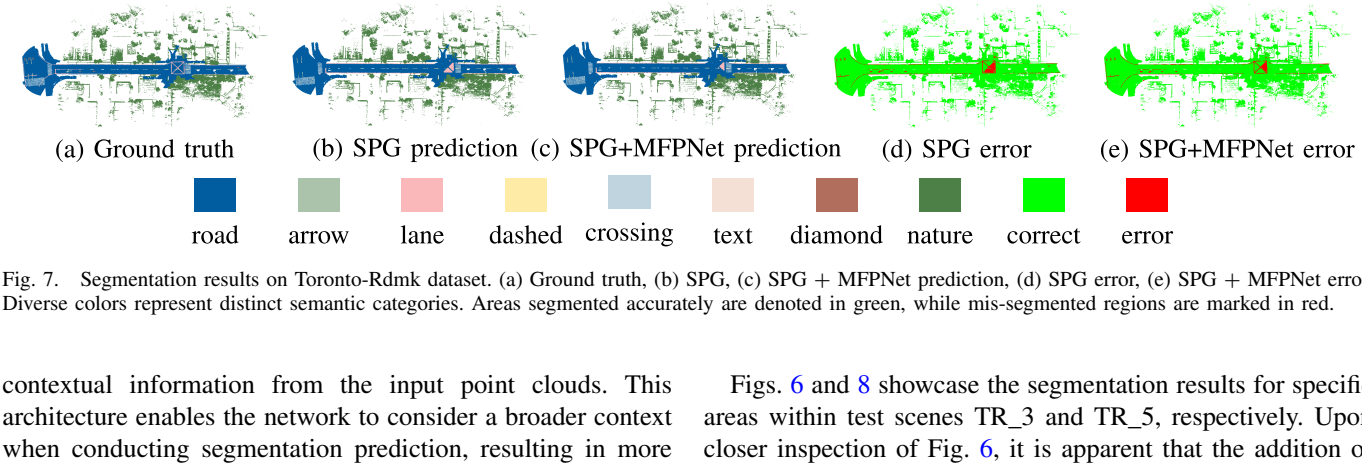

Fig. 7. Segmentation results on Toronto-Rdmk dataset. (a) Ground truth, (b) SPG, (c) SPG + MFPNet prediction, (d) SPG error, (e) SPG + MFPNet error. Diverse colors represent distinct semantic categories. Areas segmented accurately are denoted in green, while mis-segmented regions are marked in red.

contextual information from the input point clouds. This architecture enables the network to consider a broader context when conducting segmentation prediction, resulting in more accurate and coherent segmentations.

Figs. 5 and 7 depict the segmentation results for the test scenes R_3 and R_5, respectively. Each scene includes five subfigures, including (a) displaying the ground truth labels, (b) presenting the predictions generated by the SPG network, and (c) showcasing the predictions resulting from the SPG network combined with the MFPNet module. To allow for a clearer comparison of prediction discrepancies before and after the addition of the MFPNet module, all accurately segmented points are denoted in green, while all mis-segmented points are represented in red. Subfigures (d) and (e) in both Figs. 5 and 7 provide visual comparisons of the correct and erroneous points predicted by the standalone SPG network and the SPG network integrated with the MFPNet module, respectively.

Figs. 6 and 8 showcase the segmentation results for specific areas within test scenes TR_3 and TR_5, respectively. Upon closer inspection of Fig. 6, it is apparent that the addition of the MFPNet module to the SPG network leads to a reduction in segmentation errors on zebra crossings. Similarly, a detailed evaluation of Fig. 8 shows a decrease in segmentation errors on lane markings when the SPG network is supplemented with the MFPNet module. MFPNet's hierarchical architecture provides both local and global contextual information to the SPG network, enhancing its ability to comprehend point-to-point relationships within the point cloud. By integrating this information, the combined network can capture the intricacies of the data and achieve higher accuracy compared to the standalone SPG network.

4) Segmentation Results on the Toronto-3D: Table IV displays the quantitative segmentation results for several methods applied to the Toronto-3D dataset. Among these methods,

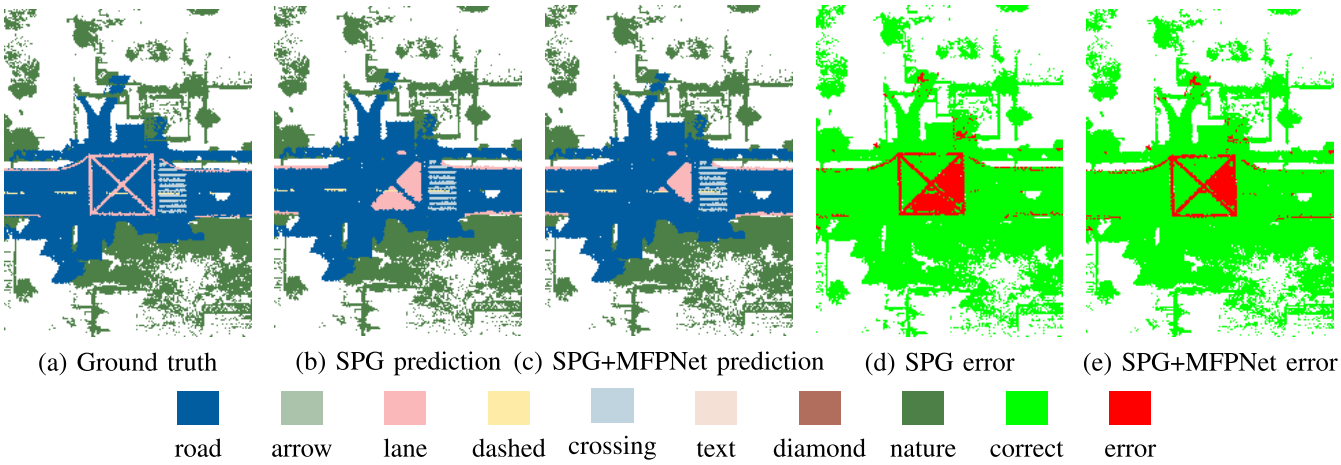

Fig. 8. Detailed segmentation results on Toronto-Rdmk dataset. (a) Ground truth, (b) SPG, (c) SPG + MFPNet prediction, (d) SPG error, (e) SPG + MFPNet error. Diverse colors represent distinct semantic categories. Areas segmented accurately are denoted in green, while mis-segmented regions are marked in red.

TABLE IV
SEGMENTATION RESULTS (%) OF DIFFERENT METHODS ON TORONTO-3D DATASET

Method	mIoU	OA	Road	Road mark.	Natural	Building	Util. line	Pole	Car	Fence
PointNet++ [44]	41.8	84.9	89.3	0.0	69.1	54.2	43.8	23.3	52.0	3.0
PointNet++ MSG [44]	59.5	92.6	92.9	0.0	86.1	82.2	61.0	62.8	76.4	14.4
TGNet [71]	61.3	94.1	93.5	0.0	90.8	81.6	65.3	63.0	88.7	7.9
DGCNN [31]	61.8	94.2	93.9	0.0	86.1	82.2	61.0	62.8	76.4	14.4
MS-PCNN [72]	65.9	90.0	93.8	3.8	93.5	82.6	67.8	72.0	91.1	22.5
KPFCNN [45]	69.1	-	94.6	0.1	96.1	91.5	**87.7**	**81.6**	85.7	15.7
MSTGNet [4]	70.5	**95.7**	94.4	**17.2**	95.7	88.8	76.0	74.0	**94.2**	23.6
KPFCNN + MFPNet	**72.6**	-	**94.7**	0.0	**96.4**	**93.1**	86.1	79.7	93.9	**36.8**

the proposed KPFCNN + MFPNet demonstrates significant advantages over the original KPFCNN [45] and other methods across multiple categories. These results indicate the effectiveness of the proposed MFPNet for segmentation improvement, especially in complex urban road scenarios. The feature pooling aggregation module and the prediction refinement module are crucial components of the MFPNet, contributing to its superior performance. The feature pooling aggregation module enhances the feature representation by incorporating contextual information and improving feature discriminability. Meanwhile, the prediction refinement module improves fine-grained localization, corrects errors, and enhances robustness to noise. These factors collectively contribute to the improved segmentation results and higher IoU scores achieved by KPFCNN + MFPNet compared to other models. When comparing the class-specific IoU scores, KPFCNN + MFPNet demonstrates superiority in several categories. Notably, it achieves remarkable performance for building (93.1%), car (93.9%), and fence (36.8%), which is substantially better than the original KPFCNN that achieves 91.5% for building, 85.7% for car, and 15.7% for fence. i

B. Road Marking Classification

1) Dataset Description: We employed the RdmkNet dataset, which encompasses six distinct categories: arrow, lane, dashed, crossing, text, and diamond. The dataset has been systematically partitioned into a training set with 10,287 samples and a test set containing 4,413 samples. Moreover, we augmented the training process on the RdmkNet dataset by incorporating noise and normals. To maintain consistency and ensure fair comparison, we uniformly segregated the training and test sets and upheld parameter standardization throughout the training phase.

2) Classification Results on the RdmkNet: We compared three methods, namely PT [73], PCT [74], and SortNet [75], aiming to adapt the transformer architecture for 3D point clouds. Each method introduced unique self-attention mechanisms tailored to capture local and global contextual information. Despite sharing a common goal, they proposed different networks and experimental setups. Remarkably, these methods demonstrated exceptional performance in various tasks, e.g., point cloud classification and segmentation, outperforming state-of-the-art methods on many benchmark datasets.

We evaluated these methods for road marking classification using two common criteria: instance accuracy and class accuracy. Instance accuracy measures the percentage of instances that are correctly classified, while class accuracy measures the percentage of classes that are correctly classified. Table V displays the classification results on the RdmkNet dataset. The results demonstrated that PCT outperformed the other methods in terms of both instance accuracy and class accuracy, with scores of 92.3% and 88.6%, respectively. SortNet also achieved relatively high scores, but it fell behind PCT.

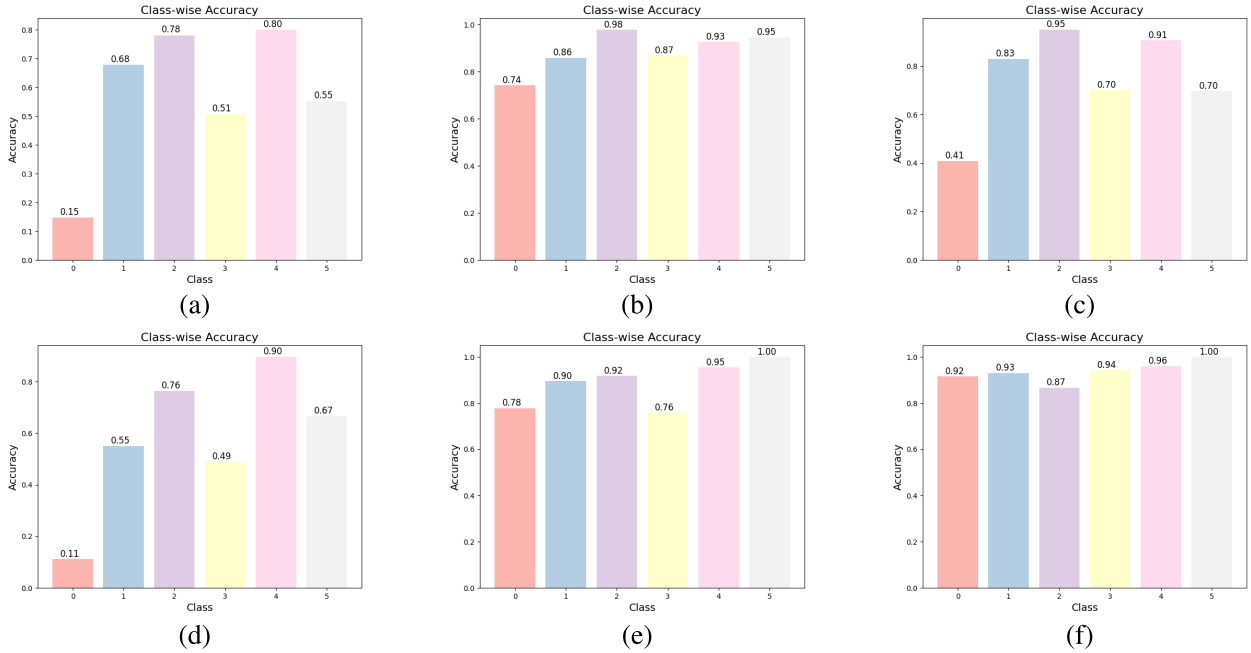

Fig. 9. Comparison of class accuracies for PT, PCT, and SortNet networks on the original RdmkNet dataset (a-c) and its extended version with added normals and noise (d-f) across six classes. Each subfigure represents the class accuracy for a specific network on the respective dataset.

Additionally, PT indicated the lowest scores among these three methods, with instance accuracy and class accuracy scores of 71.9% and 57.5%, respectively.

Table VI shows the classification results of the different transformer-based methods on the RdmkNet dataset with noise and normals. The results demonstrate that SortNet outperforms the other methods with the highest instance accuracy and class accuracy of 92.1% and 93.5%, respectively. PCT and PT show similar classification performance, with instance accuracy and class accuracy of 91.6% and 88.3%, and 73.1% and 57.8%, respectively.

TABLE V

CLASSIFICATION RESULTS (%) OF DIFFERENT METHODS ON THE RDMKNET DATASET

Method	Instance Accuracy	Class Accuracy
PT [73]	71.9	57.5
PCT [74]	92.3	88.6
SortNet [75]	86.7	74.9

TABLE VI

CLASSIFICATION RESULTS (%) OF DIFFERENT METHODS ON THE RDMKNET DATASET WITH NOISE AND NORMALS

Method	Instance Accuracy	Class Accuracy
PT [73]	73.1	57.8
PCT [74]	91.6	88.3
SortNet [75]	92.1	93.5

C. Ablation Study

We evaluated the performance of three different deep learning networks, PT, PCT, and SortNet, on the RdmkNet dataset and their extended versions. These extended versions include added normals and noise and cover six different classes. We evaluated the performance based on various metrics, including class accuracy and confusion matrix.

Fig. 9 shows a comparison of class accuracies achieved by each network on both versions of the dataset. The results indicate that the performance of all networks was affected by the addition of noise and normals. Specifically, SortNet and PCT demonstrated stronger robustness and adaptability, as most categories showed significant improvements in accuracy after incorporating noise and normals. In contrast, the PT network experienced a decline in accuracy for most categories. These findings highlight the value of the RdmkNet dataset for road marking classification.

Fig. 10 shows the confusion matrices for each network on both versions of the dataset, revealing the distribution of misclassifications among different classes. These matrices can help identify which classes are frequently confused with each other. As seen in Figs. 10 (a) and (d), categories labeled as 2 (dashed line) and 4 (lane line) are the most easily confused. In the original RdmkNet dataset, 294 instances of the dashed line category were predicted as lane lines, while 228 instances of the lane line category were predicted as dashed lines. When noise and normals were introduced in the RdmkNet dataset, 353 instances of the dashed line category were predicted as lane lines and 111 instances of the lane line category were predicted as dashed lines. Moreover, as shown in Figs. 10 (f), 195 instances of the dashed line category were predicted as lane lines. These results confirmed the discriminative training capabilities of the released RdmkNet dataset with or without the added noise and normals. The RdmkNet dataset showcases its advantages in providing diverse and challenging scenarios that aid in evaluating the networks' ability to differentiate between similar classes. The confusion

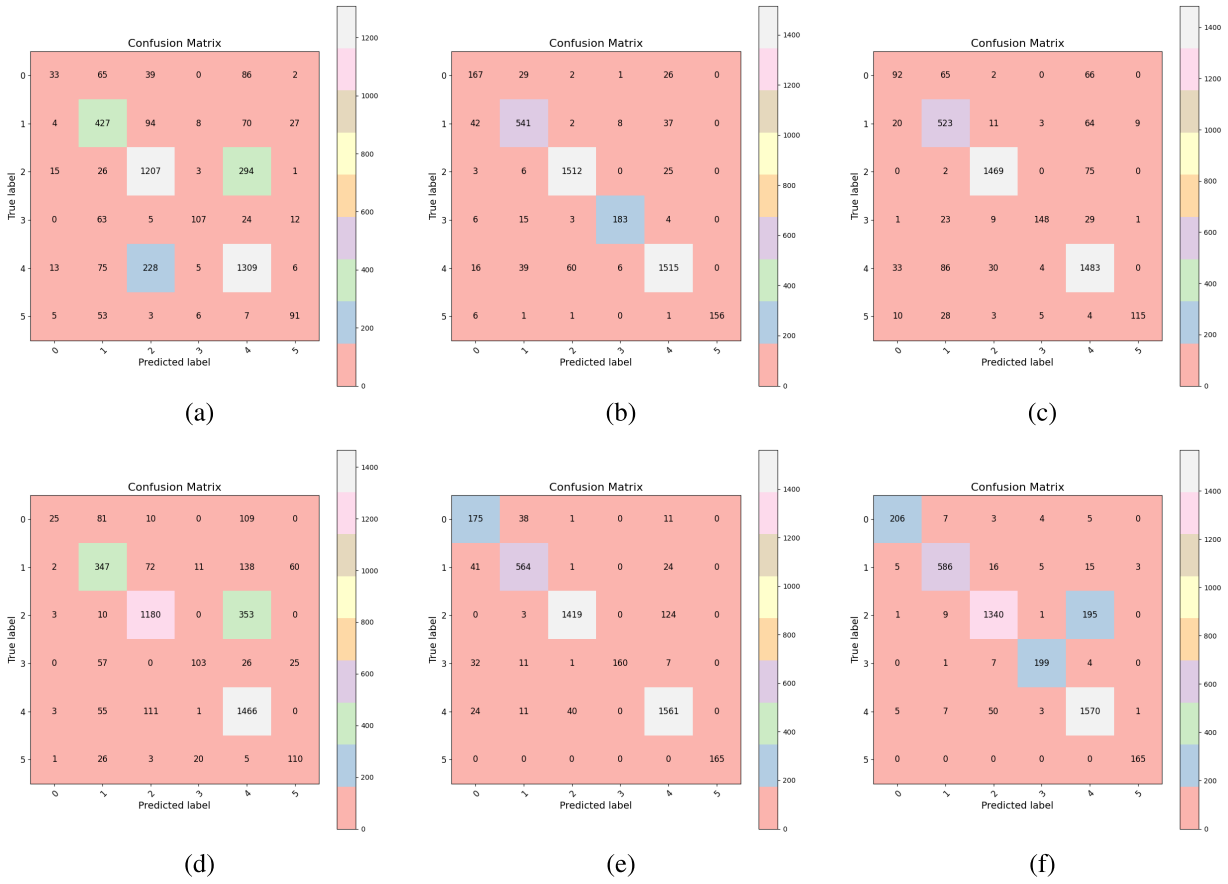

Fig. 10. Comparison of confusion matrices for PT, PCT, and SortNet networks on the original RdmkNet dataset (a-c) and its extended version with added normals and noise (d-f) across six classes. Each subfigure represents the confusion matrix for a specific network on the respective dataset.

between the dashed line and lane line categories highlights the dataset's complexity and real-world applicability, which is crucial in developing more effective and robust models for 3D point cloud processing.

VI. Conclusion

This study presents two point cloud benchmarks, RdmkNet and Toronto-Rdmk, tailored for the classification and segmentation of road markings within complex urban scenarios. Alongside these benchmarks, we introduce MFPNet, an innovative multi-level feature optimization network structure. MFPNet is segmented into three pivotal components: the M-Transformer module, the feature pooling aggregation module, and the prediction refinement module. The M-Transformer module effectively captures rich contextual information and long-range dependencies within point clouds. This is complemented by the feature pooling aggregation module, which strategically employs max-pooling to extract salient features and average-pooling for a comprehensive global feature understanding. The prediction refinement module addresses the nuances of MFPNet by emphasizing and incorporating the more subtle, often overlooked point relationships with lower attention scores, ensuring a refined predictive capability for MFPNet. The benchmarks, RdmkNet and Toronto-Rdmk, coupled with the robustness of MFPNet, represent a significant progression in road marking classification and segmentation, contributing to advancements in the autonomous driving domain.

Acknowledgment

The authors would like to express their gratitude to the Spatial Sensing and Computing Laboratory at Xiamen University for their support in providing the raw road marking point cloud data.

References

[1] Y. He et al., "Deep learning based 3D segmentation: A survey," 2021, *arXiv:2103.05423*.

[2] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3D point clouds: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 12, pp. 4338–4364, Dec. 2021.

[3] M. Liu, Y. Zhou, C. R. Qi, B. Gong, H. Su, and D. Anguelov, "Less: Label-efficient semantic segmentation for LiDAR point clouds," in *Proc. Eur. Conf. Comput. Vis.*, vol. 13699, 2022, pp. 70–89.

[4] W. Tan et al., "Toronto-3D: A large-scale mobile LiDAR dataset for semantic segmentation of urban roadways," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 797–806.

[5] O. Unal, D. Dai, and L. Van Gool, "Scribble-supervised LiDAR semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2687–2697.

[6] Q. Hu et al., "SQN: Weakly-supervised semantic segmentation of large-scale 3D point clouds," in *Proc. Eur. Conf. Comput. Vis.*, vol. 13687, 2022, pp. 600–619.

[7] I. Armeni et al., "3D semantic parsing of large-scale indoor spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1534–1543.

[8] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2432–2443.

[9] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3D.net: A new large-scale point cloud classification benchmark," 2017, *arXiv:1704.03847*.

[10] X. Roynard, J.-E. Deschaud, and F. Goulette, "Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification," *Int. J. Robot. Res.*, vol. 37, no. 6, pp. 545–557, May 2018.

[11] J. Behley et al., "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9296–9306.

[12] Z. Wu et al., "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.

[13] C. Wen, X. Sun, J. Li, C. Wang, Y. Guo, and A. Habib, "A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 147, pp. 178–192, Jan. 2019.

[14] Y. Li et al., "Deep learning for LiDAR point clouds in autonomous driving: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 8, pp. 3412–3432, Aug. 2021.

[15] M. Cheng, H. Zhang, C. Wang, and J. Li, "Extraction and classification of road markings using mobile laser scanning point clouds," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 3, pp. 1182–1196, Mar. 2017.

[16] R. Yang, Q. Li, J. Tan, S. Li, and X. Chen, "Accurate road marking detection from noisy point clouds acquired by low-cost mobile LiDAR systems," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 10, p. 608, Oct. 2020.

[17] L. Ma et al., "Capsule-based networks for road marking extraction and classification from mobile LiDAR point clouds," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 4, pp. 1981–1995, Apr. 2021.

[18] S. Chen, Z. Zhang, R. Zhong, L. Zhang, H. Ma, and L. Liu, "A dense feature pyramid network-based deep learning model for road marking instance segmentation using MLS point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 784–800, Jan. 2021.

[19] L. Fang, T. Sun, S. Wang, H. Fan, and J. Li, "A graph attention network for road marking classification from mobile LiDAR point clouds," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 108, Apr. 2022, Art. no. 102735.

[20] O. Iparraguirre, N. Iturbe-Olleta, A. Brazalez, and D. Borro, "Road marking damage detection based on deep learning for infrastructure evaluation in emerging autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 22378–22385, Nov. 2022.

[21] K. Peng et al., "MASS: Multi-attentional semantic segmentation of LiDAR data for dense top-view understanding," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15824–15840, Sep. 2022.

[22] T. Yu, J. Meng, and J. Yuan, "Multi-view harmonized bilinear network for 3D object recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 186–194.

[23] Z. Yang and L. Wang, "Learning relationships for multi-view 3D object recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7504–7513.

[24] C. R. Qi, H. Su, M. NieBner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view CNNs for object classification on 3D data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5648–5656.

[25] W. Wang, H. Zhou, G. Chen, and X. Wang, "Fusion of a static and dynamic convolutional neural network for multiview 3D point cloud classification," *Remote Sens.*, vol. 14, no. 9, p. 1996, Apr. 2022.

[26] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.

[27] Y. Ben-Shabat, M. Lindenbaum, and A. Fischer, "3DmFV: Three-dimensional point cloud classification in real-time using convolutional neural networks," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3145–3152, Oct. 2018.

[28] A. S. Gezawa, Z. A. Bello, Q. Wang, and L. Yunqi, "A voxelized point clouds representation for object classification and segmentation on 3D data," *J. Supercomput.*, vol. 78, no. 1, pp. 1479–1500, Jan. 2022.

[29] H. Zhao, L. Jiang, C.-W. Fu, and J. Jia, "PointWeb: Enhancing local neighborhood features for point cloud processing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5565–5573.

[30] Y. Liu, B. Fan, S. Xiang, and C. Pan, "Relation-shape convolutional neural network for point cloud analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8895–8904.

[31] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 146:1–146:12, Oct. 2019.

[32] R. Hassan, M. M. Fraz, A. Rajput, and M. Shahzad, "Residual learning with annularly convolutional neural networks for classification and segmentation of 3D point clouds," *Neurocomputing*, vol. 526, pp. 96–108, Mar. 2023.

[33] F. J. Lawin, M. Danelljan, P. Tosteberg, G. Bhat, F. S. Khan, and M. Felsberg, "Deep projective 3D semantic segmentation," *Proc. CAIP*, vol. 2, pp. 95–107, 2017.

[34] A. Boulch, B. L. Saux, and N. Audebert, "Unstructured point cloud semantic labeling using deep segmentation networks," in *Proc. EG3DOR*, 2017, pp. 1–8.

[35] B. Wu, X. Zhou, S. Zhao, X. Yue, and K. Keutzer, "SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 4376–4382.

[36] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "RangeNet++: Fast and accurate LiDAR semantic segmentation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 4213–4220.

[37] Y. Guo and T. Chen, "Semantic segmentation of RGBD images based on deep depth regression," *Pattern Recognit. Lett.*, vol. 109, pp. 55–64, Jul. 2018.

[38] J. Huang and S. You, "Point cloud labeling using 3D convolutional neural network," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 2670–2675.

[39] A. Dai, D. Ritchie, M. Bokeloh, S. Reed, J. Sturm, and M. Nießner, "ScanComplete: Large-scale scene completion and semantic segmentation for 3D scans," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4578–4587.

[40] B. Graham, M. Engelcke, and L. van der Maaten, "3D semantic segmentation with submanifold sparse convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9224–9232.

[41] C. Choy, J. Gwak, and S. Savarese, "4D spatio-temporal ConvNets: Minkowski convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3075–3084.

[42] H. Su et al., "SPLATNet: Sparse lattice networks for point cloud processing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2530–2539.

[43] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.

[44] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 5099–5108.

[45] H. Thomas, C. R. Qi, J. Deschaud, B. Marcotegui, F. Goulette, and L. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6410–6419.

[46] L. Wang, Y. Huang, Y. Hou, S. Zhang, and J. Shan, "Graph attention convolution for point cloud semantic segmentation," in *Proc. IEEE CVPR*, Jun. 2019, pp. 10296–10305.

[47] Z. Zeng, Y. Xu, Z. Xie, W. Tang, J. Wan, and W. Wu, "LEARD-Net: Semantic segmentation for large-scale point cloud scene," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 112, Aug. 2022, Art. no. 102953.

[48] F. Liu et al., "3DCNN-DQN-RNN: A deep reinforcement learning framework for semantic parsing of large-scale 3D point clouds," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5679–5688.

[49] A. Dai and M. Nießner, "3DMV: Joint 3D-multi-view prediction for 3D semantic scene segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 11214, 2018, pp. 458–474.

[50] M. Jaritz, J. Gu, and H. Su, "Multi-view PointNet for 3D scene understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 3995–4003.

[51] H.-Y. Chiang, Y.-L. Lin, Y.-C. Liu, and W. H. Hsu, "A unified point-based framework for 3D segmentation," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2019, pp. 155–163.

[52] X. Zhu et al., "Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9939–9948.

[53] Y. Hou, X. Zhu, Y. Ma, C. C. Loy, and Y. Li, "Point-to-voxel knowledge distillation for LiDAR semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8469–8478.

[54] S. Fan, Q. Dong, F. Zhu, Y. Lv, P. Ye, and F.-Y. Wang, "SCF-Net: Learning spatial contextual features for large-scale point cloud segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14504–14513.

[55] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inform. Process. Syst. (NIPS)*, 2017, pp. 5998–6008.

[56] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 2978–2988.

[57] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1. MI, USA: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[58] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 5754–5764.

[59] H. Hu, Z. Zhang, Z. Xie, and S. Lin, "Local relation networks for image recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3463–3472.

[60] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. 9th ICLR*, 2021, pp. 1–21.

[61] H. Zhao, J. Jia, and V. Koltun, "Exploring self-attention for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10073–10082.

[62] L. Fan et al., "Embracing single stride 3D object detector with sparse transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8448–8458.

[63] X. Lai et al., "Stratified transformer for 3D point cloud segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8490–8499.

[64] W. Zhao, W. Wang, and Y. Tian, "GraFormer: Graph-oriented transformer for 3D pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 20406–20415.

[65] J. Wang, Y. Cui, D. Guo, J. Li, Q. Liu, and C. Shen, "PointAttN: You only need attention for point cloud completion," *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 6, pp. 5472–5480, Mar. 2024.

[66] L. Landrieu and M. Simonovsky, "Large-scale point cloud semantic segmentation with superpoint graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4558–4567.

[67] Y. Lin et al., "FPConv: Learning local flattening for point convolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4292–4301.

[68] M. Xu, R. Ding, H. Zhao, and X. Qi, "PAConv: Position adaptive convolution with dynamic kernel assembling on point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3173–3182.

[69] X. Wu., Y. Lao, L. Jiang, X. Liu, and H. Zhao, "Point transformer V2: Grouped vector attention and partition-based pooling," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 33330–33342.

[70] H. Hu, F. Wang, Z. Zhang, Y. Wang, L. Hu, and Y. Zhang, "GAM: Gradient attention module of optimization for point clouds analysis," in *Proc. AAAI*, 2023, pp. 835–843.

[71] Y. Li, L. Ma, Z. Zhong, D. Cao, and J. Li, "TGNet: Geometric graph CNN on 3-D point cloud segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3588–3600, May 2020.

[72] L. Ma, Y. Li, J. Li, W. Tan, Y. Yu, and M. A. Chapman, "Multi-scale point-wise convolutional neural networks for 3D object segmentation from LiDAR point clouds in large-scale environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 821–836, Feb. 2021.

[73] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 16239–16248.

[74] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "PCT: Point cloud transformer," *Comput. Vis. Media*, vol. 7, no. 2, pp. 187–199, Jun. 2021.

[75] N. Engel, V. Belagiannis, and K. Dietmayer, "Point transformer," *IEEE Access*, vol. 9, pp. 134826–134840, 2021.

Jing Du received the M.Sc. degree from Jimei University, Xiamen, China, in 2022. She is currently pursuing the Ph.D. degree with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON, Canada. Her research interests include point cloud analysis and processing, aiming to explore high precision, and low memory consumption methods for semantic segmentation of 3D point clouds.

Lingfei Ma (Member, IEEE) received the M.Sc. and Ph.D. degrees in geomatics engineering from the University of Waterloo, Waterloo, ON, Canada, in 2017 and 2020, respectively.

He is currently an Associate Professor with the Central University of Finance and Economics, Beijing, China. He has published more than 30 papers in refereed journals and conferences, including IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, *ISPRS Journal of Photogrammetry and Remote Sensing*, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and IEEE-CVPRW. His research interests include autonomous driving, mobile laser scanning, intelligent processing of point clouds, 3D scene modeling, and machine learning. He was a recipient of the 2020 National Best Ph.D. Thesis Award granted by Canadian Remote Sensing Society. He serves as the Editorial Board Member for the *International Journal of Applied Earth Observation and Geoinformation*.

Jing Li received the M.Sc. degree in geographic information systems and remote sensing from Beijing Forestry University, Beijing, China, in 2017, and the Ph.D. degree from the Department of Information Technology and Cyber Security, People's Public Security University of China, Beijing, in 2021. He is currently an Associate Professor with the School of Information, Central University of Finance and Economics. His research interests include remote sensing, image processing, and pattern recognition.

Nannan Qin received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2019. He is currently a Lecturer of remote sensing and geomatics engineering with Nanjing University of Information Science and Technology, Nanjing, China. His main research interests include LiDAR point clouds, 3-D vision, and GeoAI.

John Zelek (Member, IEEE) received the Ph.D. degree from McGill University, Quebec, Canada, in 1996. He was formerly the Associate Graduate Chair of the Systems Design Engineering from 2013 to 2017. He is currently an Associate Professor and the Co-Director of the Vision Image Processing (VIP) Laboratory, University of Waterloo, Waterloo, ON, Canada. His main research interests include autonomous robotic mapping and localization, 3D scene understanding, man made infrastructure assessment (e.g., roads, buildings, and bridges), eye (fundus, OCT) image understanding for disease, learning 3D models from single-views, and athletic sport tracking and biomechanical understanding of play and ability from video feeds.

Haiyan Guan (Senior Member, IEEE) received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2009, and the Ph.D. degree in geomatics from the University of Waterloo, Waterloo, ON, Canada, in 2014.

She is currently a Professor with the School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, China. She has published more than 50 research papers in refereed journals, books, and proceedings, including IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, *ISPRS Journal of Photogrammetry and Remote Sensing*, and IGARSS and ISPRS proceedings. Her current research interests include information extraction from LiDAR point clouds and from earth observation images.

Jonathan Li (Fellow, IEEE) received the Ph.D. degree in geomatics engineering from the University of Cape Town, South Africa, in 2000. He is currently a Professor of geomatics and systems design engineering with the University of Waterloo, Canada. He has coauthored almost 600 publications, more than 150 of which were published in top remote sensing journals, including *Remote Sensing of Environment*, *ISPRS Journal of Photogrammetry and Remote Sensing*, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and *International Journal of Applied Earth Observation and Geoinformation* (*Journal of Applied Gerontology*). He has also published papers in flagship conferences in computer vision and AI, including CVPR, AAAI, and IJCAI. He has supervised nearly 200 master's/Ph.D. students as well as post-doctoral fellows/visiting scholars to completion. His main research interests include AI-based information extraction from earth observation images and LiDAR point clouds, pointgrammetry and remote sensing, GeoAI and 3D vision for digital twin cities, and autonomous driving. He is a fellow of Canadian Academy of Engineering, the Royal Society of Canada (Academy of Science), and the Engineering Institute of Canada. He is the President of Canadian Institute of Geomatics (CIG). He is the Editor-in-Chief of JAG and an Associate Editor of IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING and IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS.