UDTIRI: An Online Open-Source Intelligent Road Inspection Benchmark Suite

Sicen Guo, *Graduate Student Member, IEEE*, Jiahang Li, *Graduate Student Member, IEEE*, Yi Feng, Dacheng Zhou, Denghuang Zhang, Chen Chen, Shuai Su, *Student Member, IEEE*, Xingyi Zhu, Qijun Chen, *Senior Member, IEEE*, and Rui Fan, *Senior Member, IEEE*

***Abstract*— In the emerging field of urban digital twins (UDTs), there are extensive and captivating opportunities for leveraging cutting-edge deep learning techniques. Particularly within the specialized area of intelligent road inspection (IRI), a noticeable gap exists, underscored by the current dearth of dedicated research efforts and the lack of large-scale well-annotated datasets. To foster advancements in this burgeoning field, we have launched an online open-source benchmark suite, referred to as UDTIRI. Along with this article, we introduce the road pothole detection task, the first online competition published within this benchmark suite. This task provides a well-annotated dataset, comprising 1,000 RGB images and their pixel/instance-level ground-truth annotations, captured in diverse real-world scenarios under different illumination and weather conditions. Our benchmark provides a systematic and thorough evaluation of state-of-the-art object detection, semantic segmentation, and instance segmentation networks, developed based on either convolutional neural networks or Transformers. We anticipate that our benchmark suite will serve as a catalyst for the integration of advanced UDT techniques into IRI. By providing algorithms with a more comprehensive understanding of diverse road conditions, we seek to unlock their untapped potential and foster innovation in this critical domain.**

***Index Terms*— Urban digital twins, deep learning, intelligent road inspection, benchmark suite, road pothole detection.**

I. Introduction

DIGITAL twin (DT) represents the forefront of technology, including innovative algorithms that bridge physical systems with digital networks [1]. This integration blurs the boundaries between the physical and digital domains, heralding a new era of interconnectedness and real-time analytics [2]. With the rapid pace of digital transformation, the applications of DT technology expand across diverse sectors, from smart manufacturing [3] to intelligent urban planning [4] and advanced medical healthcare [5]. An urban digital twin (UDT) is fashioned by encoding the semantic and geospatial properties of urban entities, such as buildings and roads [6]. These digital replicas of physical urban infrastructures are indispensable to fulfill a diverse range of needs and uses, as exemplified by applications such as intelligent road inspection (IRI) [7].

Traditional road inspection is typically conducted by structural engineers or certified inspectors [8]. However, this process is fraught with challenges: it is perilous, inefficient, costly, and tedious [9]. Additionally, the road inspection results are often qualitative and subjective, relying solely on the expertise of the individual inspectors [10]. With the advancement of UDT techniques, especially deep neural networks, there is an increasing appetite for data-driven IRI systems, which generally undertake two primary tasks [11]: (1) road data acquisition [12] and (2) road damage detection [13]. Developing a comprehensive, open-source, and well-annotated online benchmark suite for evaluating UDT techniques applied to IRI is, therefore, of paramount significance to the intelligent transportation society.

Potholes, among the most prevalent types of road damage, are considerably large structural defects on the road surface [14]. Detecting these defects is not only vital for proactive urban road maintenance but also imperative for autonomous driving [15]. However, current autonomous driving perception systems prioritize the detection of large objects of interest, *e.g.*, pedestrians, traffic signs, and vehicles, often sidelining

Manuscript received 11 April 2023; revised 27 August 2023 and 2 January 2024; accepted 3 January 2024. This work was supported in part by the National Key R&D Program of China under Grant 2023YFE0202400, in part by the National Natural Science Foundation of China under Grant 62233013 and Grant 52278455, in part by the Science and Technology Commission of Shanghai Municipal under Grant 22511104500, in part by the Shuguang Program of Shanghai Education Development Foundation and Shanghai Municipal Education Commission under Grant 21SG24, in part by the International Cooperation Project of Science and Technology Commission of Shanghai Municipality under Grant 22210710700, in part by the Fundamental Research Funds for the Central Universities, and Xiaomi Young Talents Program. The Associate Editor for this article was Z. Ma. *(Sicen Guo and Jiahang Li are co-first authors.) (Corresponding author: Rui Fan.)*

Sicen Guo, Jiahang Li, Yi Feng, and Rui Fan are with the Machine Intelligence and Autonomous Systems (MIAS) Group, the College of Electronics and Information Engineering, Shanghai Research Institute for Intelligent Autonomous Systems, the State Key Laboratory of Intelligent Autonomous Systems, and the Frontiers Science Center for Intelligent Autonomous Systems, Tongji University, Shanghai 201804, China (e-mail: guosicen@tongji.edu.cn; lijiahang617@tongji.edu.cn; fengyi@ieee.org; rui.fan@ieee.org).

Dacheng Zhou, Denghuang Zhang, and Chen Chen are with the Department of Control Science and Engineering, Tongji University, Shanghai 201804, China (e-mail: zhoudacheng20@tongji.edu.cn; zhangdenghuang666@gmail.com; ccsama0109@gmail.com).

Shuai Su and Qijun Chen are with the Robotics and Artificial Intelligence Laboratory (RAIL), College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China (e-mail: sushuai@tongji.edu.cn; qjchen@tongji.edu.cn).

Xingyi Zhu is with the Department of Road and Airport Engineering and the Key Laboratory of Road and Traffic Engineering of Ministry of Education, Tongji University, Shanghai 200092, China (e-mail: zhuxingyi66@tongji.edu.cn).

Data is available on-line at https://udtiri.com.

Digital Object Identifier 10.1109/TITS.2024.3351209

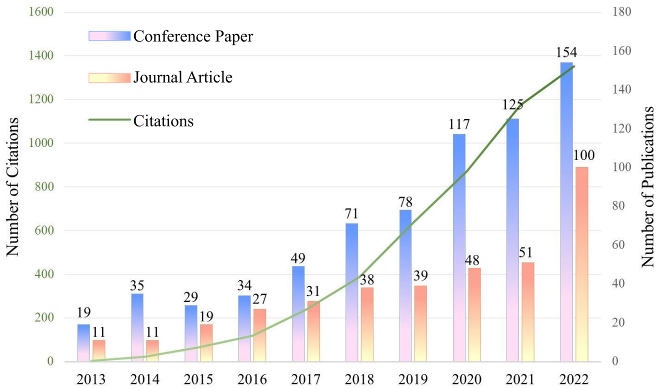

Fig. 1. Publication and citation trends for road pothole detection over the past decade. Conference papers are sourced from the Engineering Village database (webpage: engineeringvillage.com), while journal articles and citations are sourced from the Web of Science database (webpage: webofscience.com).

road damage. Nevertheless, driving quality, vehicle maneuverability, fuel consumption, and tire longevity are all related to road conditions. Therefore, accurate and efficient detection of road potholes is crucial for improving both driving comfort and safety [7].

Fig. 1 highlights the growing interest in road pothole detection over the past decade, affirming its position as a burgeoning research topic. Our recent survey article [11] categorizes the existing road pothole detection algorithms into three groups: (1) classical 2-D image processing-based, (2) 3-D point cloud modeling and segmentation-based, and (3) data-driven approaches. The first category of algorithms generally utilizes explicit image processing algorithms to segment road RGB or disparity/depth images [16]. Such algorithms are often computationally demanding and sensitive to various environmental factors, notably illumination and weather conditions [17]. Additionally, the irregular shapes of road potholes render the geometric assumptions made in such approaches occasionally infeasible. Therefore, 3-D point cloud modeling and segmentation-based algorithms have become popular choices for road pothole detection [18]. These algorithms typically consider the 3-D road point clouds, captured using a range sensor, as a quadratic surface. The raw point clouds are then segmented by comparing the differences between the modeled surface and the actual data [19]. However, these algorithms are still relatively underutilized. This is primarily because accurate 3-D road imaging is costly, and real-world road surfaces can be highly irregular and uneven, sometimes rendering these techniques impractical. Data-driven approaches, typically developed based on convolutional neural networks (CNNs), have emerged as frontrunners, delivering compelling road pothole detection results [9], [15], [20], [21].

Over the past decade, the advent of several online benchmark suites, such as KITTI [22] and Cityscapes [23], has been playing a pivotal role in advancing the performance of general visual perception algorithms. However, despite the abundance of such datasets for general computer vision research, the specific domain of IRI, especially when underpinned by cutting-edge UDT techniques, remains relatively underexplored. A primary reason is that road defects such as potholes are not ubiquitous, making the creation of large-scale datasets inherently challenging [14]. Furthermore, most existing approaches in this area simply apply transfer learning to fine-tune state-of-the-art (SoTA) object detection or semantic segmentation models on relatively small road inspection datasets. Moreover, previous studies in this research area typically reported results based on experiments where datasets were split randomly. Comparing algorithms on the same dataset without consistent data splits can skew results, as performance may be influenced by overlapping training and validation data distributions. Finally, while all existing road pothole detection datasets are created for either instance-level object detection or pixel-level semantic segmentation, there is a noticeable absence of large-scale datasets designed to accommodate both tasks simultaneously via instance segmentation. Therefore, developing an online open-source benchmark suite comprising a variety of IRI tasks, including but not limited to road surface 3D reconstruction and road damage detection, is a popular area of research that requires more attention.

In this article, we introduce the **U**rban **D**igital **T**wins for **I**ntelligent **R**oad **I**nspection (**UDTIRI**) online benchmark suite, accessible at https://udtiri.com. Road pothole detection, the first online competition launched within this benchmark suite, provides researchers with a large-scale, well-annotated dataset for comprehensive evaluation of object detection, semantic segmentation, and instance segmentation networks, designed for this specific task. Similar to KITTI [22] and Cityscapes [23], the ground-truth annotations (see Fig. 2) are available for model training and validation, while the evaluation metrics on the test set can be acquired by uploading results to the UDTIRI benchmark suite. To set a reference point, we have conducted extensive experiments with 14 SoTA object detection networks, 30 SoTA semantic segmentation networks, and 10 SoTA instance segmentation networks, providing baseline results for road pothole detection. With additional online competitions launched within this benchmark suite in the near future, we believe that it will serve as a catalyst for the integration of cutting-edge UDT methodologies into IRI.

The remainder of this article is structured as follows: Sect. II presents a comprehensive review of SoTA object detection, semantic segmentation, and instance segmentation networks. Sect. III details our UDTIRI benchmark suite and road pothole detection dataset. Sect. IV presents the conducted experiments and provides both qualitative and quantitative comparisons of the SoTA networks. Sect. V discusses the potential limitations of the compared networks. Finally, we summarize our contributions in Sect. VI.

II. Literature Review

A. Object Detection Networks

Existing CNN-based object detection methods are predominantly categorized into two groups: two-stage and one-stage approaches. Two-stage methods first generate regions of interest (RoIs), which are then refined for both object classification and bounding box regression. Nevertheless, the sequential nature of two-stage methods can result in slower inference speeds. On the other hand, one-stage methods formulate

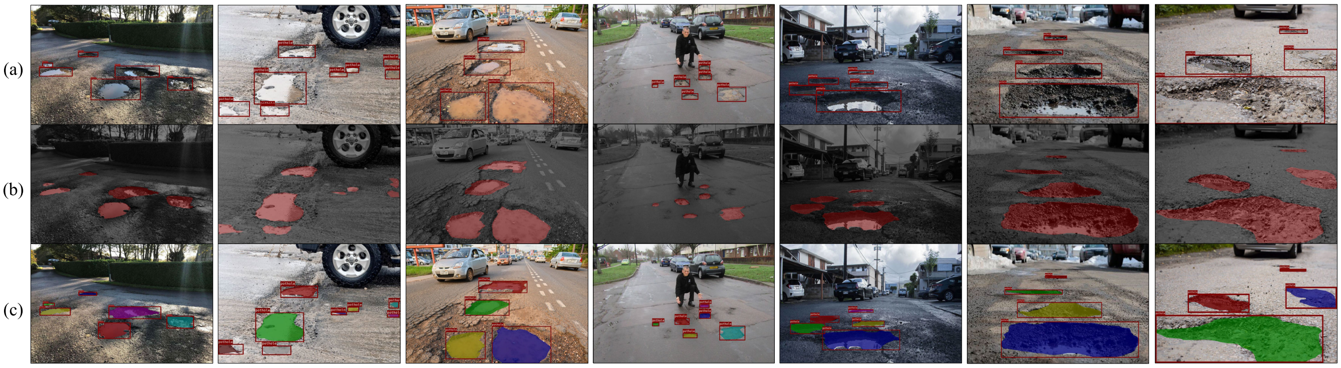

Fig. 2. Examples of the ground-truth annotations for the road pothole detection competition within the UDTIRI benchmark suite: (a) object detection; (b) semantic segmentation; (c) instance segmentation.

object detection as a direct bounding box regression task, typically achieving higher computational efficiency yet sometimes sacrificing detection accuracy compared to two-stage methods.

As a pioneering two-stage approach, R-CNN [24] introduces region proposals, leverages a CNN to extract features from these proposals, and classifies these proposals with a support vector machine (SVM). Unfortunately, its multi-stage training pipeline is computationally intensive. To address this limitation, Fast R-CNN [25] streamlines the object detection process by pooling CNN features associated with each region proposal, significantly improving overall efficiency by sharing computations for overlapping regions. However, it still relies on the relatively slow selective search for region proposal generation. Therefore, Faster R-CNN [26] addresses this drawback by introducing a region proposal network (RPN), which directly generates region proposals, thereby enabling end-to-end training.

One-stage methods, exemplified by the you only look once (YOLO) series, single-shot multi-box detector (SSD) [27], CenterNet [28], RetinaNet [29], and EfficientDet [30], have gained increasing attention owing to their remarkable real-time performance. These approaches can be broadly categorized as anchor-free or anchor-based. The fundamental difference between these two categories lies in their reliance on pre-defined anchors to assist object detection. Anchor-based object detection frameworks require careful anchor design to adequately capture the scale and aspect ratio of specific object classes. In contrast, anchor-free approaches forgo the use of pre-defined anchors and instead directly predict a bounding box for each object.

As the first anchor-free approach, YOLOv1 [31] simplifies object detection into a direct bounding box regression task. It utilizes a CNN for feature extraction and a fully connected layer to regress object bounding box coordinates and classes. Furthermore, YOLOv6 [32] incorporates the SCYLLA-IoU (SIoU) [33] bounding box regression loss, which leads to improved object detection accuracy when compared to earlier anchor-based models, such as YOLOv2 through YOLOv5. Additionally, CenterNet [28] represents each object as a single point at the bounding box center, instead of regressing the entire bounding box. This point-based representation generally enhances model generalizability and can be readily extended to related tasks, such as 3D object detection, instance segmentation, and keypoint estimation.

Among the anchor-based YOLO series, YOLOv2 [34] employs a high-resolution classifier and anchor boxes to improve object detection accuracy. Nevertheless, YOLOv2 struggles with fine-grained localizations, overlapping objects, and potential loss of spatial information, even though it achieves real-time performance. YOLOv3 [35] addresses these limitations by making predictions across three different scales, capturing more comprehensive semantics to improve object detection performance. YOLOv4 [36] incorporates a cross-stage partial network [37] into its backbone, notably enhancing learning capability while also reducing computational complexity compared to YOLOv2 and YOLOv3. Moreover, YOLOv5 [38] utilizes an embedded anchor box selection mechanism to improve training and inference speed compared to YOLOv4. YOLOv7 [39] introduces an extended efficient layer aggregation network to further boost inference speed. It achieves the most favorable trade-off between efficiency and accuracy when compared to all previous versions of YOLO. SSD [27] was also developed with the primary aim of achieving a balance between speed and accuracy by utilizing pre-defined anchors and multi-scale features. Additionally, EfficientDet [30] optimizes the trade-off among model complexity, speed, and accuracy by adjusting network depth, width, and input resolution. Unfortunately, most one-stage detectors still lag behind two-stage models in accuracy, primarily due to sensitivity to foreground-background class imbalances. RetinaNet [29] addresses this limitation by employing focal loss to focus on "hard" samples, allowing it to maintain high-speed processing while remaining competitive with SoTA one-stage methods.

DETR [40] employs an encoder-decoder Transformer architecture along with a set-based global loss to produce an optimal bipartite matching between predicted and ground-truth objects. This loss function uniquely associates each prediction with a specific target object, ensuring invariance to the order of predictions. Inspired by the deformable convolution, Deformable DETR [41] incorporates sparse spatial sampling to overcome the challenges of slow training convergence and high computational complexity inherent in DETR. Specifically, it utilizes a deformable attention module to focus on a small set of key sampling points around a reference point, regardless of the spatial size of the feature maps. This addresses a limitation in the standard Transformer attention mechanism, which typically considers all possible spatial locations and thus converges slowly during training.

B. Semantic Segmentation Networks

Fully convolutional network (FCN) [42] marked a pioneering milestone in the use of CNNs for end-to-end semantic segmentation. However, its segmentation does not fully account for pixel relationships, resulting in segmentation results that lack spatial consistency. Additionally, FCN also significantly amplifies memory usage and computational complexity. To address these limitations, Fast FCN [43] extracts high-resolution feature maps through upsampling convolutions. This approach effectively addresses the spatial inconsistency issue and significantly reduces computational complexity by more than threefold.

Recent approaches [44], [45], [46], [47], [48], [49] have made significant strides in enhancing performance by expanding the receptive fields using pyramid-based multi-resolution techniques. Pyramid scene parsing network (PSPNet) [49] performs spatial pyramid pooling (SPP) at multiple scales, achieving exceptional performance across several semantic segmentation benchmarks. Similarly, based on Mask R-CNN [50] and feature pyramid network (FPN) [51], panoptic FPN [44] utilizes a lightweight semantic segmentation branch for dense pixel prediction. Furthermore, DeepLabv3 [46] employs several parallel atrous SPP (ASPP) modules to gather contextual information across multiple scales. Nevertheless, the stride operations used in DeepLabv3 may lead to the loss of object boundary details. To address this limitation, DeepLabv3+ [47] introduces a concise yet effective decoder into DeepLabv3, significantly improving semantic segmentation results, particularly along label boundaries. Additionally, dynamic multi-scale network (DMNet) [48] learns variable-scale features through dynamic multi-scale filters. It is more adaptable and flexible, as each branch can capture a unique scale of features relevant to the input image.

U-Net [52] features a U-shaped encoder-decoder structure, originally designed for biomedical image segmentation problems. In contrast to symmetric encoder-decoder architectures utilized in the following studies [52], [53], [54], efficient neural network (ENet) [55] adopts a larger encoder paired with a smaller decoder. The encoder effectively handles data with lower resolutions, thereby providing the decoder with fine-grained features. Subsequently, the decoder samples these features and refines boundary details. SegResNet [56], another asymmetric encoder-decoder architecture, replaces the encoder of SegNet [54] with ResNet blocks. Moreover, it incorporates a variational autoencoder branch to regularize the shared encoder by reconstructing input images. Unlike these prior arts [42], [52], [53], [54], [55], [56] that focus on the recovery of high-resolution feature maps from low-resolution representations, high-resolution network (HRNet) [57] maintains high-resolution representations throughout the entire feature extraction and fusion process, resulting in more accurate predictions, achieved through progressive and repetitive multi-scale feature fusion, conducted by multi-resolution subnetworks in parallel.

Attention mechanisms have been playing a pivotal role in recent semantic segmentation networks [58], [59]. As two notable approaches, the non-local neural network (Non-local) [60] obtains the attention mask by computing the correlation matrix between each point in the feature maps, and PSANet [61] learns the mask by aggregating context information for each specific point in a self-adaptive manner. However, the extensive computational demands of attention mechanisms have limited their application in various real-world scenarios. To overcome this challenge, the asymmetric non-local neural network (ANN) [62] samples only a few representative points from the feature maps, significantly reducing computational complexity. Additionally, the attention computation can be decomposed into a pair-wise term and a unary term, which can be challenging to learn independently. The disentangled non-local network (DNLNet) [63] addresses this issue by decoupling the tight relationship between these two components. Most attention-based approaches [61], [64] use adaptive weights to compute pair-wise similarity or learn pixel-wise attention maps. However, they tend to overlook the importance of global guidance from the feature extractors. To address this limitation, adaptive pyramid context network (APCNet) [65] estimates the degree of sub-region contribution from local and global representations and leverages multi-scale representations with a feature pyramid, resulting in improved overall performance.

While attention mechanisms have demonstrated superior performance compared to ASPP [46], large convolutional kernels, and stacked convolutional layers, their heightened demand for GPU memory can often be prohibitively expensive. Therefore, several networks have emerged with a primary focus on further minimizing these computational requirements. Criss-cross network (CCNet) [66] introduces fully spatial attention, while interlaced sparse self-attention network (ISANet) [67] factorizes the dense affinity matrix into the product of two sparse affinity matrices. Furthermore, instead of treating all pixels as reconstruction bases [60], [61], the expectation maximization attention network (EMANet) [68] finds a more compact basis set, leading to a substantial reduction in computational complexity. Additionally, context encoding network (ENCNet) [69] selectively highlights the class-dependent feature maps, thereby infusing the scene-relevant prior information into the network. This technique simplifies the generation of large attention maps while notably reducing memory consumption.

Vision Transformer (ViT) [70] has been gaining momentum in recent years. Swin Transformer [71], Segmenter [72], and Twins [73] are all developed based on ViT [70]. Swin Transformer designs a hierarchical Transformer architecture that computes representations with shifted windows. Inspired by DETR [40], Segmenter develops a mask Transformer decoder, capable of capturing global context at each layer during both encoding and decoding stages. To improve semantic segmentation at both global and local scales, Twins [73] adopts a two-branch architecture: one captures global contextual information, while the other one focuses on the local boundary details of the segmented regions. Furthermore, SegFormer [74] aggregates information from various layers, effectively combining both local and global attention to produce robust representations. To improve the learned representations, ResNeSt [75] combines channel-wise attention with multi-path representation into a single unified split-attention block. Similar to the self-attention mechanism used in ViT, object-contextual representation (OCR) [76] characterizes pixels by exploiting the representations of corresponding object classes. The conventional multi-scale context schemes, such as SPP [77] and ASPP [46], only differentiate pixels with different spatial positions, while OCR [76] distinguishes between contextual pixels of the same object class and those of different object classes.

C. Instance Segmentation Networks

Similar to object detection networks, instance segmentation networks can also be broadly categorized as either two-stage and one-stage ones [78]. The former networks [50], [79], [80], [81], [82] first detect bounding boxes for each instance and then perform pixel classification within each bounding box to generate the final mask. In contrast, one-stage networks [83], [84], [85], [86] directly propose prediction boxes from the input images without a region proposal step [87]. Although one-stage methods are more suitable for applications requiring real-time performance due to their concise and efficient architectures, two-stage methods typically achieve higher segmentation accuracy, primarily attributed to the second refined stage [88].

Mask R-CNN [50] is a pioneering two-stage framework that extends Faster R-CNN [26] by adding an additional branch to predict pixel-level masks in parallel with the existing branch that recognizes bounding boxes. Its architecture consists of a CNN backbone, a RPN, and two heads separately for object classification and prediction. The introduction of the RoI align (abbreviated as RoIAlign) module ensures that the extracted features are correctly aligned, thus eliminating the misalignment issues caused by quantization errors present in previous methods [26]. Cascade R-CNN [79] is another extension of Faster R-CNN [26], which improves instance detection accuracy through a cascade of multiple stages. Unlike the methods [26], [50], [79] that usually predict mask quality score based on the confidence of instance segmentation networks, Mask Scoring R-CNN [80] takes both the instance feature and the corresponding predicted mask into account to regress the intersection over union (IoU) score for masks. This approach considers the accuracy of both semantic categories and the instance masks, presenting a novel method for scoring the instance segmentation quality and offering a new perspective on the evaluation of instance segmentation performance.

Unlike Mask R-CNN [50] that relies on RoI operations (typically RoIAlign) to obtain the final instance masks, YOLACT [81] decouples RoI detection from the feature maps used for mask prediction. Additionally, instead of using instance-wise RoIs as inputs to a network with fixed weights, CondInst [89] employs dynamic instance-aware networks conditioned on instances. This approach offers two notable advantages: (1) it eliminates the need for RoI cropping and feature alignment through an FCN module; (2) with the increased capacity of dynamically generated conditional convolutions, the compact mask head leads to significantly faster inference speed. The same research group also proposed BoxInst [82], which realizes instance segmentation with the use of two losses: (1) a surrogate loss that focuses on minimizing the discrepancy between the projections of the ground-truth box and the predicted mask, and (2) a pair-wise loss that supervises the label consistency in proximal pixels, determining whether two pixels have the same labels or not.

As a representative one-stage instance segmentation network, segmenting objects by locations (SOLO) [83] uses a similar paradigm to semantic segmentation for the instance segmentation task. Basically, the mask branch predicts soft masks for all potential objects, while the category branch subsequently determines the object classes, enabling efficient instance segmentation without utilizing RoI operations. Nonetheless, SOLO struggles to segment small instances. To overcome this limitation, SOLOv2 [84] dynamically predicts mask kernels based on the input and assigns appropriate location categories to different pixels. Additionally, to prevent duplicate predictions, it employs "matrix non-maximum suppression (NMS)", which dramatically boost the model's inference speed.

Capturing contextual information and long-range dependencies is crucial for instance segmentation. Global context network (GCNet) [85] simplifies the non-local network [60] by explicitly utilizing a query-independent attention map applicable to all query positions. GCNet has proven to deliver impressive performance, primarily due to its capacity to model pixel-level long-range dependencies while simultaneously mapping channel-wise attention. Additionally, to address the challenge of capturing long-range dependencies, deformable convolutional network (DCN) [90] introduces deformable convolution, which offers dynamic and learnable receptive fields, effectively adapting to the image content. It solves the inherent limitations of geometric transformations in CNNs and has demonstrated exceptional performance across various computer vision tasks.

III. UDTIRI Benchmark Suite

Our initial aim is centered around the creation of an online open-source benchmark suite, designed to offer comprehensive evaluations of cutting-edge UDT techniques applied to tackle IRI problems. These evaluations involve various general visual perception algorithms, including but not limited to object detection (instance-level perception), semantic segmentation (pixel-level perception), and instance segmentation (perception at both instance and pixel levels) networks. However, it is worth noting that these general models have rarely been evaluated specifically for IRI tasks, primarily due to the lack of a public dataset that includes diverse forms of ground-truth annotations and provides reasonable data partitions. Therefore in this paper, we take the initial step by launching a road pothole detection competition based on a large-scale, well-annotated, multi-purpose, real-world dataset.

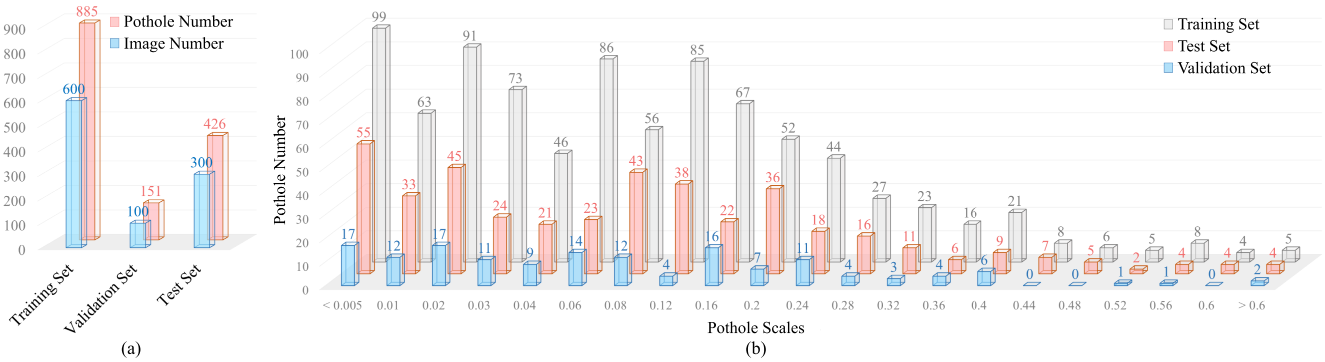

Fig. 3. Dataset characteristics: (a) a histogram showing the pothole distribution; (b) a histogram showing the distribution of pothole scales.

Prior to introducing our newly developed road pothole detection dataset, we first provide a brief overview of the relevant existing datasets that have been created for the evaluation of visual perception algorithms. Labeling object detection ground truth is relatively inexpensive and can result in larger datasets. For example, a dataset[1] [91] was created for object detection-based pothole detection, which contains 3,777 RGB images for training and 628 RGB images for testing. However, in the context of road inspection, our primary focus is on acquiring accurate information (*e.g.*, shapes and sizes) of road potholes. This objective inherently requires accurate pixel-level annotations. In our previous work [13], we published the first pixel-level road pothole detection dataset,[2] which contains 67 collections of RGB images (resolution: $800 \times 1,312$ pixels), subpixel disparity images, transformed disparity images, and ground-truth annotations. Furthermore, we published a relatively larger dataset, referred to as the Pothole-600 dataset[3] [15], which contains 600 pairs of RGB images and transformed disparity images. While the datasets mentioned above are suitable for the evaluation of semantic segmentation algorithms, they were created under rather limited illumination and weather conditions. Moreover, the road potholes in these datasets are comparatively easy to recognize from such scenarios.

To address the absence of large-scale, multi-functional, well-annotated road pothole datasets, we collect road data with respect to diverse pothole depths, sizes, and shapes, captured using different cameras mounted on different vehicles and under various weather and illumination conditions. These comprehensive data are not only suitable for model training in each of these individual tasks but can also be leveraged for multi-task learning, thereby setting our dataset apart from existing options. Within our dataset, the road potholes have a wide range of scales, as depicted in Fig. 3. Based on these statistical analyses, we categorize the road potholes into large, medium, and small ones. In our experiments, we conduct a comprehensive performance evaluation of object detection and instance segmentation networks with respect to the different scales of road potholes.

All the images in our dataset have been annotated and are available in various formats. For object detection, we utilize the VOC [92] and COCO [93] formats. In the case of semantic segmentation, the VOC format is employed, while for instance segmentation, we utilize the COCO format. We have made our training and validation sets, along with ground-truth annotations, publicly available. To evaluate the performance of algorithms on our test set, researchers can submit their results via our online benchmark suite. This feature serves dual purposes: it not only streamlines the process of comparing and validating various algorithms but also cultivates a collaborative environment within the research community, promoting the exchange of methods and results. Through its automated and standardized evaluation mechanism, we believe that our benchmark suite represents a vital advancement in this field.

IV. Experimental Results

A. Experimental Setup

We conduct extensive baseline experiments using 14 object detection networks, 30 semantic segmentation networks, and 10 instance segmentation networks. All networks are implemented in PyTorch. All experiments are conducted on an NVIDIA RTX 3090 GPU and an Intel Xeon Platinum 8255C CPU. Each network is trained for 150 epochs. We keep the default settings of each network. The quantitative results of object detection, semantic segmentation, and instance segmentation are presented in Tables I, II, and III, respectively. Additionally, the qualitative experimental results of object detection, semantic segmentation, and instance segmentation are shown in Figs. 4, 5, and 6, respectively.

B. Evaluation Metrics

We use average precision (AP) and mean IoU (mIoU) as the evaluation metrics in the object detection task. Furthermore, we utilize accuracy (Acc), IoU, precision (Pre), recall (Rec), and F1-score (Fsc) to quantify the performance of semantic segmentation networks. Moreover, we use AP as the evaluation metric in the instance segmentation task.

Following the evaluation on the MS COCO [93] dataset, we compute AP with respect to different sizes of road potholes: small (denoted as AP_S), medium (denoted as AP_M), and large (denoted as AP_L). We use two proportion thresholds to determine small, medium, and large road potholes. Referring to Fig. 3, the first 300 potholes are considered small (with a pothole area proportion of less than 1.12%). The potholes

[1] kaggle.com/sovitrath/road-pothole-images-for-pothole-detection

[2] github.com/ruirangerfan/stereo_pothole_datasets

[3] sites.google.com/view/pothole-600

TABLE I
QUANTITATIVE COMPARISON OF OBJECT DETECTION NETWORKS, WITH THE BEST RESULTS SHOWN IN BOLD TYPE

Network	Params	FPS	Validation Set					Test Set				
			AP_S (%) ↑	AP_M (%) ↑	AP_L (%) ↑	AP_A (%) ↑	mIoU (%) ↑	AP_S (%) ↑	AP_M (%) ↑	AP_L (%) ↑	AP_A (%) ↑	mIoU (%) ↑
YOLOv1 [31]	28.49 M	84.36	13.40	32.10	45.10	35.10	54.70	10.70	25.20	45.10	33.60	53.90
YOLOv2 [34]	28.52 M	23.67	14.30	39.90	53.80	42.10	60.50	11.20	28.10	46.10	34.50	57.60
YOLOv3 [35]	61.52 M	55.33	24.20	44.80	55.90	47.40	69.70	26.60	43.00	56.00	47.50	66.00
YOLOv4 [36]	63.94 M	36.05	31.10	47.60	66.80	56.60	76.70	31.30	49.00	61.30	52.60	74.30
YOLOv5 [38]	46.14 M	36.84	42.70	60.20	74.10	65.30	76.60	40.70	56.50	67.60	59.80	73.60
YOLOv6 [32]	58.47 M	48.73	48.60	**66.10**	77.10	**69.60**	83.90	52.20	**66.60**	**76.40**	**69.60**	**82.90**
YOLOv7 [39]	37.20 M	49.20	45.30	59.70	73.90	65.50	82.70	44.90	57.90	74.90	65.40	78.80
Faster R-CNN [26]	136.69 M	25.29	21.40	52.30	64.40	55.00	77.00	21.30	54.00	63.50	53.30	75.30
RetinaNet [29]	36.33 M	34.78	25.90	51.40	58.20	50.50	58.00	22.70	40.70	54.90	45.50	56.30
CenterNet [28]	191.24 M	23.38	27.80	53.40	66.30	55.90	75.60	29.50	52.70	64.30	55.10	74.30
SSD [27]	23.61 M	136.73	29.10	54.80	69.90	59.30	79.90	30.80	53.90	70.00	59.10	78.10
EfficientDet [30]	8.01 M	12.89	41.20	54.50	74.20	63.60	79.60	40.60	57.40	73.30	64.10	79.00
DETR [40]	41.30 M	30.80	43.20	62.40	**80.50**	68.90	**85.90**	45.30	61.00	74.40	65.40	81.30
Deformable DETR [41]	39.85 M	31.60	**50.20**	63.50	79.60	69.50	82.60	**56.40**	61.50	75.00	68.10	80.60

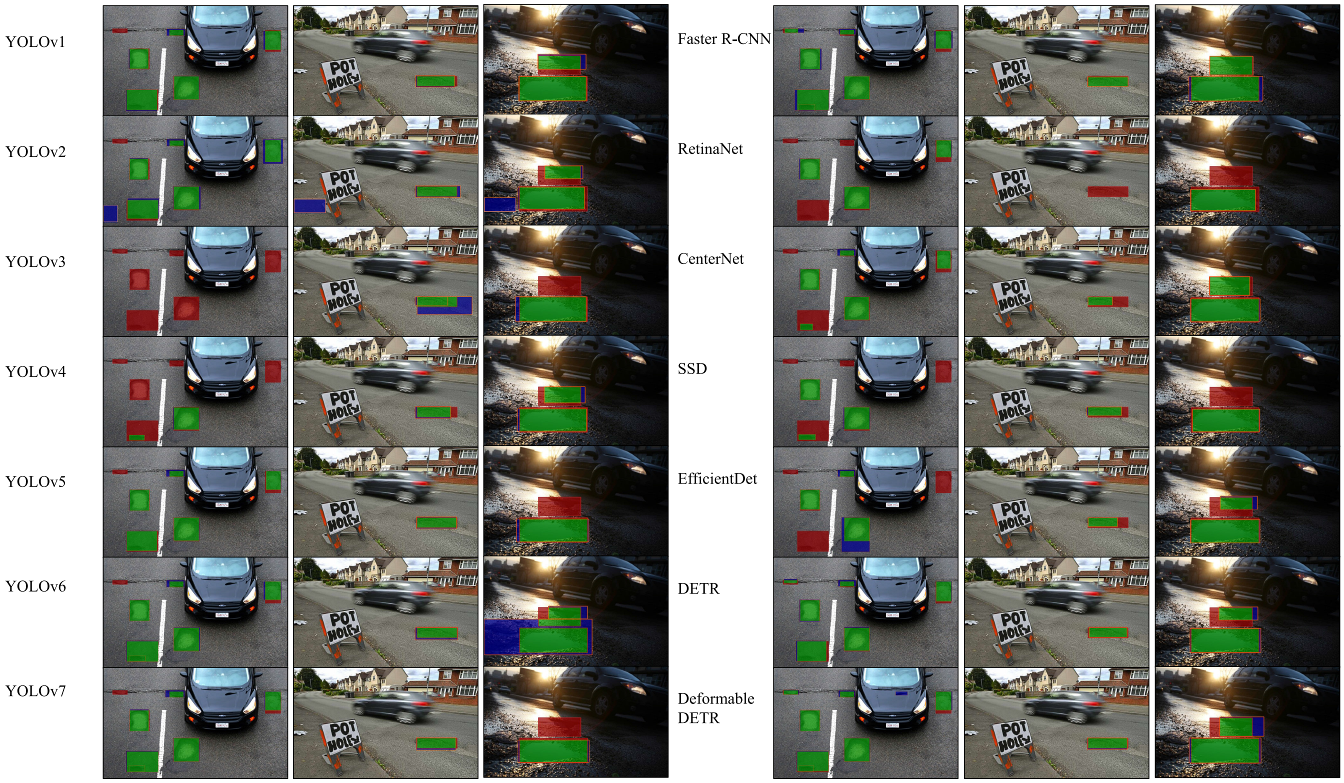

Fig. 4. Qualitative experimental results of object detection. The green areas in the image represent true-positive predictions, the blue areas represent false-positive predictions, and the red areas represent false-negative predictions.

numbered from 301 to 600 (with a pothole area proportion between 1.12% and 3.72%) are considered medium, while the remaining potholes are considered large (with a pothole area proportion greater than 3.72%). Additionally, AP_A represents an average AP score that provides a comprehensive evaluation of the model's performance across all pothole sizes.

C. Object Detection Network Performance

The quantitative results obtained from the object detection networks, as outlined in Table I, reveal the following insights: (1) YOLOv6 achieves the highest AP_A and AP_M, and Deformable DETR achieves the highest AP_S on both the validation and test sets; (2) regarding AP_L and mIoU, DETR outperforms others on the validation set, while YOLOv6 leads with the highest scores on the test set; (3) YOLOv6 demonstrates real-time performance, and is slightly faster than DETR and Deformable DETR.

Although most networks are capable of detecting larger potholes, they often struggle with medium and smaller ones. Transformer-based networks, such as DETR and Deformable DETR, consistently outperform CNN-based networks in detecting small potholes. This superiority can be attributed to the fact that small potholes might become undetectable after passing through multiple convolution layers in CNN-based models. In contrast, Transformer-based approaches, with a self-attention mechanism, can effectively capture information from all positions without relying on pooling operations, allowing them to preserve important details relevant to small potholes. However, the generalizability of Transformer-based networks is limited, likely due to data availability constraints. While these networks show promise for small object

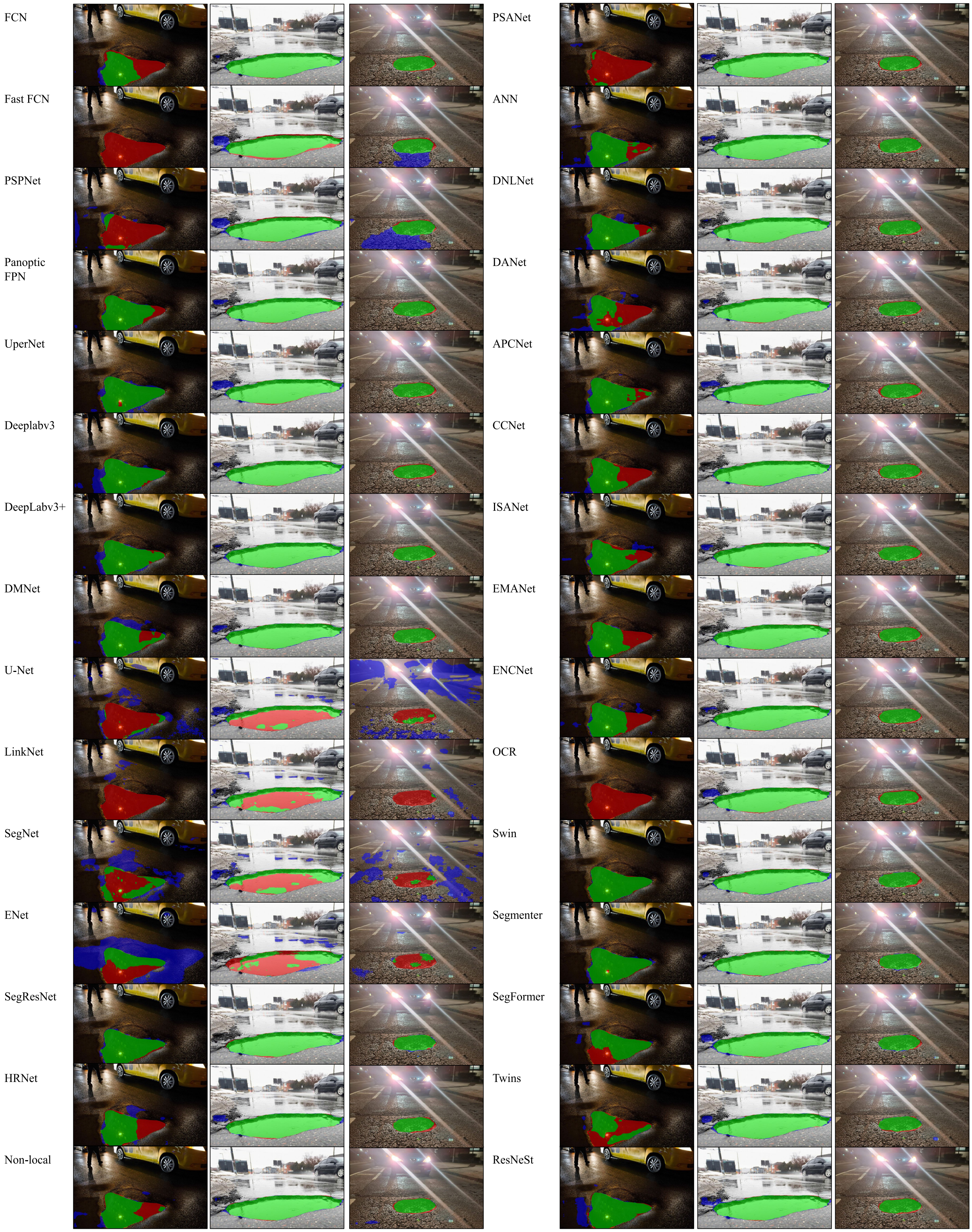

Fig. 5. Qualitative experimental results of semantic segmentation. The green areas in the image represent true-positive predictions, the blue areas represent false-positive predictions, and the red areas represent false-negative predictions.

detection, further research and data collection efforts may be necessary to enhance their performance across diverse scenarios.

It is also worth noting that YOLOv6 demonstrates a higher occurrence of false-positive regions during night time, as illustrated in Fig. 4. This indicates its potential limitations

TABLE II
QUANTITATIVE COMPARISON OF SEMANTIC SEGMENTATION NETWORKS, WITH THE BEST RESULTS SHOWN IN BOLD TYPE

Network	Params	FPS	Validation Set					Test Set				
			IoU (%) ↑	Fsc (%) ↑	Pre (%) ↑	Rec (%) ↑	Acc (%) ↑	IoU (%) ↑	Fsc (%) ↑	Pre (%) ↑	Rec (%) ↑	Acc (%) ↑
FCN [42]	49.48 M	16.54	79.60	88.64	91.81	85.69	85.69	74.96	85.96	87.39	84.05	97.25
Fast FCN [43]	87.85 M	10.47	81.28	89.67	92.33	87.17	96.93	76.63	86.77	88.57	85.04	97.47
PSPNet [49]	48.98 M	16.14	81.90	90.05	92.23	87.96	97.34	74.93	85.67	85.05	86.30	97.18
Panoptic FPN [44]	28.51 M	18.46	77.14	87.10	90.91	83.59	96.61	73.23	84.55	85.14	83.69	97.07
UperNet [45]	41.40 M	17.20	79.97	88.87	89.97	87.81	96.99	73.72	84.87	84.51	85.24	96.98
DeepLabv3 [46]	68.11 M	13.07	79.79	88.76	91.81	85.90	97.02	76.86	86.90	87.75	86.10	97.43
DeepLabv3+ [47]	43.58 M	16.15	81.16	89.60	89.64	89.57	97.16	75.88	86.28	84.52	88.13	97.28
DMNet [48]	53.28 M	14.52	79.82	88.78	92.63	85.23	97.05	74.39	85.32	89.11	81.83	97.23
U-Net [52]	29.06 M	12.10	69.42	81.92	83.35	80.62	94.57	64.12	78.15	78.74	78.13	94.41
LinkNet [53]	44.00 M	11.70	76.23	86.51	86.42	86.57	96.14	72.83	84.24	84.25	84.27	95.72
SegNet [54]	212.96 M	1.23	71.62	83.44	84.73	82.21	95.11	66.93	80.10	81.32	79.06	94.42
ENet [55]	1.36 M	21.98	74.23	85.10	86.21	83.93	95.70	74.91	85.62	86.95	84.38	96.03
SegResNet [56]	204.31 M	2.45	81.91	90.14	90.91	82.65	97.13	79.25	88.34	88.53	88.29	96.82
HRNet [57]	65.86 M	22.46	80.46	89.17	91.77	86.73	97.12	76.63	86.77	88.57	85.04	97.73
Non-local [60]	50.03 M	14.13	80.82	89.40	89.07	89.72	97.09	71.99	83.72	83.12	84.32	96.74
PSANet [61]	48.98 M	12.57	80.67	89.30	93.10	85.80	97.19	76.43	86.64	86.91	86.37	97.36
ANN [62]	46.23 M	15.18	80.91	89.44	88.50	90.41	97.08	76.70	86.82	87.94	85.72	97.43
DNLNet [63]	50.13 M	10.15	79.36	88.49	90.95	86.16	96.93	73.26	84.57	86.75	82.49	97.01
DANet [64]	49.82 M	14.37	74.47	85.37	91.03	80.37	96.23	73.19	84.52	85.75	83.33	97.01
APCNet [65]	56.46 M	14.16	81.05	89.53	**94.06**	85.42	97.27	75.23	85.87	88.66	83.24	97.27
CCNet [66]	49.83 M	15.70	80.40	89.13	92.43	86.07	97.13	75.93	86.32	89.27	83.56	97.36
ISANet [67]	37.69 M	18.85	80.60	89.26	91.24	87.19	97.13	76.15	86.46	88.99	84.07	97.40
EMANet [68]	42.09 M	18.17	80.71	89.32	93.16	85.79	97.19	75.02	85.73	**90.01**	81.83	97.29
ENCNet [69]	35.89 M	17.99	79.90	88.83	91.66	86.17	97.04	74.12	85.13	84.68	85.59	97.10
OCR [76]	12.08 M	27.76	63.57	77.73	81.26	74.50	94.16	62.59	76.99	76.09	77.91	95.35
Swin [71]	121.3 M	6.55	82.70	90.53	90.45	90.16	97.41	77.98	87.63	85.04	90.37	97.47
Segmenter [72]	102.5 M	9.52	**83.21**	**90.81**	91.26	**90.41**	**97.50**	**80.74**	**89.34**	87.25	**91.54**	**97.83**
SegFormer [74]	3.75 M	15.51	73.43	84.68	87.99	81.61	95.96	74.91	85.65	87.11	84.25	97.27
Twins [73]	47.58 M	7.96	81.57	89.85	91.32	88.42	97.27	79.02	86.38	87.94	84.87	97.36
ResNeSt [75]	90.91 M	9.48	79.66	88.68	91.34	86.18	96.99	79.03	88.29	88.83	87.75	97.69

TABLE III
QUANTITATIVE COMPARISON OF INSTANCE SEGMENTATION NETWORKS, WITH THE BEST RESULTS SHOWN IN BOLD TYPE

Network	Params	FPS	Validation Set				Test Set			
			AP_S (%) ↑	AP_M (%) ↑	AP_L (%) ↑	AP_A (%) ↑	AP_S (%) ↑	AP_M (%) ↑	AP_L (%) ↑	AP_A (%) ↑
Mask R-CNN [50]	43.97 M	31.00	41.30	29.40	**61.70**	53.70	**31.00**	**55.60**	**62.40**	**52.10**
DCN [90]	97.30 M	15.20	5.10	25.10	59.30	50.10	5.00	24.20	55.10	46.30
GCNet [85]	100.00 M	12.40	15.60	**34.10**	60.90	53.60	10.10	30.30	57.20	49.50
YOLACT [81]	34.73 M	2.80	22.10	24.20	53.40	45.70	2.00	17.80	39.10	33.10
Cascade R-CNN [79]	77.02 M	24.20	33.20	33.20	61.20	**54.10**	12.70	30.60	54.80	47.70
Mask Scoring R-CNN [80]	60.23 M	30.40	**46.60**	32.00	58.80	51.60	12.20	32.80	55.30	48.60
SOLO [83]	36.12 M	30.40	0.80	15.60	52.70	42.80	1.70	18.50	45.70	37.80
SOLOv2 [84]	46.23 M	31.70	8.30	20.00	49.80	41.50	2.00	21.10	48.80	40.70
CondInst [89]	34.16 M	26.00	31.70	27.40	58.20	49.80	4.50	26.70	53.30	45.00
BoxInst [82]	34.96 M	21.70	8.90	22.60	52.80	44.40	6.10	22.50	48.20	40.60

when dealing with extreme or low-light situations. However, we still recommend YOLOv6 as the preferred choice for object detection-based road pothole detection, primarily due to its advantageous balance between speed and accuracy.

D. Semantic Segmentation Network Performance

As shown in Table II, Segmenter consistently achieves the highest IoU, Fsc, Rec, and Acc on both the validation and test sets. The results presented in Fig. 5 further illustrate that Segmenter can yield the most accurate boundaries, with minimal occurrences of false-positive regions, even under poor illumination conditions. Additionally, APCNet and EMANet stand out by achieving the highest Pre on the validation and test sets, respectively. Unfortunately, all these networks fall short in achieving real-time performance, which imposes limitations on their practical applicability in real-world scenarios.

This article represents a pioneering effort in utilizing Transformer-based networks for road pothole detection. Except for SegFormer, Transformer-based networks demonstrate superior performance compared to the CNN-based approaches. This superiority can be attributed to two key factors. First, Transformers are inherently designed to efficiently capture long-range dependencies and global context. In semantic segmentation, understanding the relationships between distant pixels or objects holds significant importance. Transformers are adept at modeling these global relationships, allowing for more context-aware segmentation. Furthermore, Transformers rely on attention mechanisms that can capture fine-grained spatial relationships within an image. This capability can lead to more precise and context-aware segmentation, particularly when dealing with densely packed objects or those with intricate shapes.

We also compare the generalizability of these networks. DeepLabv3, ENet, SegResNet, DANet, OCR, Segmenter, SegFormer, Twins, ResNeSt demonstrate comparable performance on both the validation and test sets, with the IoU fluctuating within a range of 0.06% to 2.93%. Among them, ENet achieves the best generalizability on our UDTIRI benchmark.

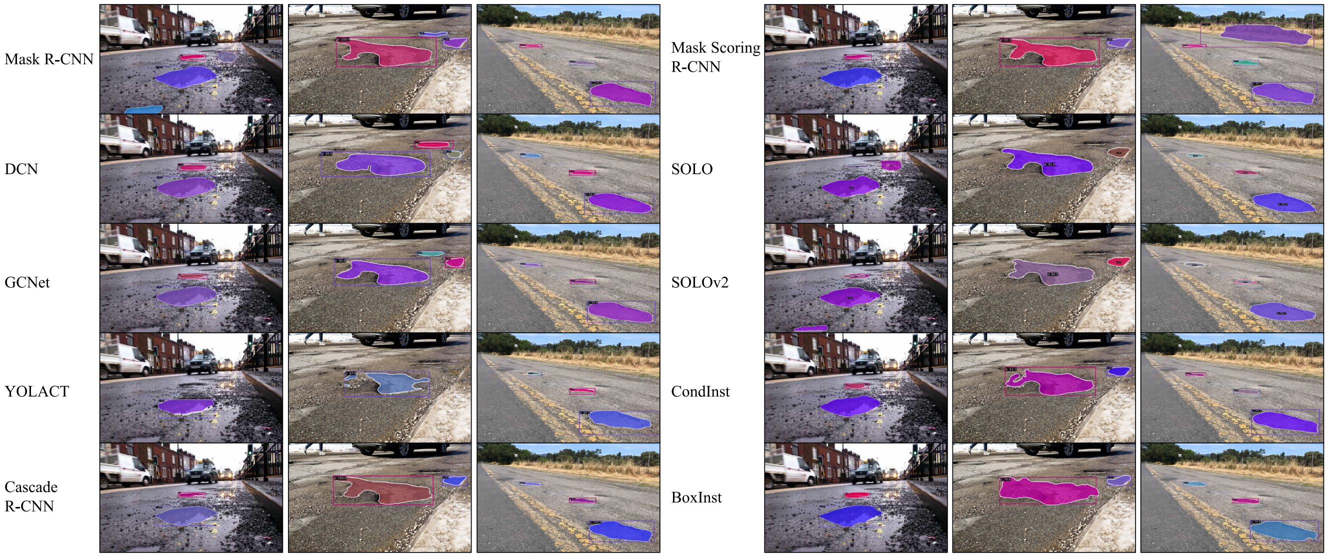

Fig. 6. Qualitative experimental results of instance segmentation. Different road potholes are shown in different colors.

E. Instance Segmentation Network Performance

The results presented in Table III reveal that Mask Scoring R-CNN, GCNet, Mask R-CNN, and Cascade R-CNN achieve the highest AP_S, AP_M, AP_L, and AP_A, on the validation set, respectively, while Mask R-CNN outperforms all other models across all evaluation metrics on the test set. These findings suggest the superior segmentation accuracy of two-stage models when compared to one-stage models. Additionally, as illustrated in Fig. 6, Mask R-CNN consistently produces accurate road pothole boundaries, even under challenging conditions characterized by high humidity and strong light reflections. Furthermore, it is worth highlighting that Mask R-CNN demonstrates comparable efficiency to one-stage approaches, such as SOLO and SOLOv2, making it a practical choice, especially for resource-limited hardware. Nonetheless, it is evident that all the instance segmentation networks compared in this study achieve unsatisfactory performance, particularly when segmenting small road potholes, underscoring the critical need for further research and improvement in this area.

V. Discussion

This study has two notable limitations. First, the current benchmark primarily focuses on single-model road pothole detection, without exploring the potential benefits of multi-sensor data fusion. Future iterations of our benchmark will incorporate additional spatial geometric information and comprehensively investigate data-fusion networks, providing a more comprehensive evaluation of model performance. Secondly, our benchmark currently focuses exclusively on potholes, omitting the inclusion of other common road damages, such as cracks. Detecting cracks is essential not only for urban road maintenance but also for automated driving perception systems. To enhance the comprehensiveness of our benchmark and align it more closely with real-world scenarios, it is crucial to incorporate additional datasets comprising various road damage types and evaluate a wider range of models for crack detection.

VI. Conclusion

In this article, we introduced an online open-source benchmark suite, referred to as UDTIRI, within which the first intelligent road inspection competition – road pothole detection was launched. The competition provides a large-scale, well-annotated dataset that can be used for the training and evaluation of object detection, semantic segmentation, and instance segmentation networks. The annotations for the training and validation sets are made accessible to researchers, where a comprehensive performance evaluation of their developed networks on the test set can be obtained by submitting the results through our online benchmark platform. Furthermore, we provided extensive baseline experimental results using 14 object detection networks, 30 semantic segmentation networks, and 10 instance segmentation networks. With upcoming IRI competitions set to be introduced within the UDTIRI benchmark, we believe that our benchmark will act as a driving force, encouraging the integration of advanced UDT techniques into IRI.

References

[1] Y. Cheng, Y. Zhang, P. Ji, W. Xu, Z. Zhou, and F. Tao, "Cyber-physical integration for moving digital factories forward towards smart manufacturing: A survey," *Int. J. Adv. Manuf. Technol.*, vol. 97, nos. 1–4, pp. 1209–1221, Jul. 2018.

[2] S. Guo, Y. Bai, M. J. Bocus, and R. Fan, "Digital transformation for intelligent road condition assessment," in *Intelligent Systems in Digital Transformation: Theory and Applications*. Cham, Switzerland: Springer, 2022, pp. 511–533.

[3] Y. H. Son, G. Y. Kim, H. C. Kim, C. Jun, and S. D. Noh, "Past, present, and future research of digital twin for smart manufacturing," *J. Comput. Des. Eng.*, vol. 9, no. 1, pp. 1–23, 2022.

[4] H. Xia, Z. Liu, M. Efremochkina, X. Liu, and C. Lin, "Study on city digital twin technologies for sustainable smart city design: A review and bibliometric analysis of geographic information system and building information modeling integration," *Sustain. Cities Soc.*, vol. 84, Sep. 2022, Art. no. 104009.

[5] M. Alazab et al., "Digital twins for Healthcare 4.0—Recent advances, architecture, and open challenges," *IEEE Consum. Electron. Mag.*, vol. 12, no. 6, pp. 29–37, 2022.

[6] B. Lei, P. Janssen, J. Stoter, and F. Biljecki, "Challenges of urban digital twins: A systematic review and a Delphi expert survey," *Autom. Construct.*, vol. 147, Mar. 2023, Art. no. 104716.

[7] R. Fan et al., "Urban digital twins for intelligent road inspection," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2022, pp. 5110–5114.

[8] R. Fan, S. Guo, and M. J. Bocus, *Autonomous Driving Perception*. Singapore: Springer, 2023.

[9] J. Li, Y. Zhang, P. Yun, G. Zhou, Q. Chen, and R. Fan, "RoadFormer: Duplex transformer for RGB-normal semantic road scene parsing," *IEEE Trans. Intell. Veh.*, 2024.

[10] S. Mathavan, K. Kamal, and M. Rahman, "A review of three-dimensional imaging technologies for pavement distress detection and measurements," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2353–2362, Oct. 2015.

[11] N. Ma et al., "Computer vision for road imaging and pothole detection: A state-of-the-art review of systems and algorithms," *Transp. Saf. Environ.*, vol. 4, no. 4, Nov. 2022, Art. no. tdac026.

[12] R. Fan, X. Ai, and N. Dahnoun, "Road surface 3D reconstruction based on dense subpixel disparity map estimation," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 3025–3035, Jun. 2018.

[13] R. Fan, U. Ozgunalp, B. Hosking, M. Liu, and I. Pitas, "Pothole detection based on disparity transformation and road surface modeling," *IEEE Trans. Image Process.*, vol. 29, pp. 897–908, 2020.

[14] R. Fan, U. Ozgunalp, Y. Wang, M. Liu, and I. Pitas, "Rethinking road surface 3-D reconstruction and pothole detection: From perspective transformation to disparity map segmentation," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 5799–5808, Jul. 2022.

[15] R. Fan, H. Wang, M. J. Bocus, and M. Liu, "We learn better road pothole detection: From attention aggregation to adversarial domain adaptation," in *Proc. Eur. Conf. Comput. Vis. Workshops*. Cham, Switzerland: Springer, Aug. 2020, pp. 285–300.

[16] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Adv. Eng. Informat.*, vol. 29, no. 2, pp. 196–210, Apr. 2015.

[17] M. R. Jahanshahi, F. Jazizadeh, S. F. Masri, and B. Becerik-Gerber, "Unsupervised approach for autonomous pavement-defect detection and quantification using an inexpensive depth sensor," *J. Comput. Civil Eng.*, vol. 27, no. 6, pp. 743–754, Nov. 2013.

[18] Z. Zhang, "Advanced stereo vision disparity calculation and obstacle analysis for intelligent vehicles," Ph.D. dissertation, Dept. Elect. Electron. Eng., Univ. Bristol, Bristol, U.K., 2013.

[19] R. Fan and M. Liu, "Road damage detection based on unsupervised disparity map segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4906–4911, Nov. 2020.

[20] A. Dhiman and R. Klette, "Pothole detection using computer vision and learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3536–3550, Aug. 2020.

[21] R. Fan, H. Wang, Y. Wang, M. Liu, and I. Pitas, "Graph attention layer evolves semantic segmentation for road pothole detection: A benchmark and algorithms," *IEEE Trans. Image Process.*, vol. 30, pp. 8144–8154, 2021.

[22] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[23] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[25] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 91–99.

[27] W. Liu et al., "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 21–37.

[28] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.

[29] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988. [Online]. Available: https://openaccess.thecvf.com/content_iccv_2017/html/Lin_Focal_Loss_for_ICCV_2017_paper.html

[30] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10778–10787.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[32] C. Li et al., "YOLOv6: A single-stage object detection framework for industrial applications," 2022, *arXiv:2209.02976*.

[33] Z. Gevorgyan, "SIoU loss: More powerful learning for bounding box regression," 2022, *arXiv:2205.12740*.

[34] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.

[35] A. Farhadi and J. Redmon, "YOLOv3: An incremental improvement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1804. Berlin, Germany: Springer, 2018, pp. 1–6.

[36] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[37] C.-Y. Wang et al., "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 390–391.

[38] G. Jocher et al., (2022), "Ultralytics/yolov5: V7.0—YOLOv5 SOTA realtime instance segmentation," *Zenodo*, doi: 10.5281/zenodo.7347926.

[39] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7464–7475. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2023/html/Wang_YOLOv7_Trainable_Bag-of-Freebies_Sets_New_State-of-the-Art_for_Real-Time_Object_Detectors_CVPR_2023_paper.html

[40] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 213–229.

[41] X. Zhu et al., "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021. [Online]. Available: https://iclr.cc/virtual/2021/oral/3448

[42] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[43] H. Wu, J. Zhang, K. Huang, K. Liang, and Y. Yu, "FastFCN: Rethinking dilated convolution in the backbone for semantic segmentation," 2019, *arXiv:1903.11816*.

[44] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6399–6408.

[45] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, "Unified perceptual parsing for scene understanding," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 418–434.

[46] L. C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[47] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.

[48] J. He, Z. Deng, and Y. Qiao, "Dynamic multi-scale filters for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3562–3572.

[49] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[50] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[51] T. Y. Lin, P. Dollàr, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.

[52] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[53] A. Chaurasia and E. Culurciello, "LinkNet: Exploiting encoder representations for efficient semantic segmentation," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.

[54] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[55] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*.

[56] A. Myronenko, "3D MRI brain tumor segmentation using autoencoder regularization," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent. Brainlesion Workshop*. Cham, Switzerland: Springer, 2018, pp. 311–320.

[57] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5693–5703.

[58] N. Jia, Y. Sun, and X. Liu, "TFGNet: Traffic salient object detection using a feature deep interaction and guidance fusion," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–11, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10225448, doi: 10.1109/TITS.2023.3293822.

[59] N. Jia, X. Liu, Y. Sun, and Z. Liu, "Enhancing IIoT vision data transmission and processing via spatial difference attention-guided saliency detection," *IEEE Internet Things J.*, p. 1, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10363628, doi: 10.1109/JIOT.2023.3343758.

[60] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[61] H. Zhao et al., "PSANet: Point-wise spatial attention network for scene parsing," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 267–283.

[62] Z. Zhu, M. Xu, S. Bai, T. Huang, and X. Bai, "Asymmetric non-local neural networks for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 593–602.

[63] M. Yin et al., "Disentangled non-local neural networks," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, Aug. 2020, pp. 191–207.

[64] J. Fu et al., "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3146–3154.

[65] J. He, Z. Deng, L. Zhou, Y. Wang, and Y. Qiao, "Adaptive pyramid context network for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7519–7528.

[66] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612.

[67] L. Huang, Y. Yuan, J. Guo, C. Zhang, X. Chen, and J. Wang, "Interlaced sparse self-attention for semantic segmentation," 2019, *arXiv:1907.12273*.

[68] X. Li, Z. Zhong, J. Wu, Y. Yang, Z. Lin, and H. Liu, "Expectation-maximization attention networks for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9167–9176.

[69] H. Zhang et al., "Context encoding for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7151–7160.

[70] A. Dosovitskiy et al., "An image is worth 16 × 16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021. [Online]. Available: https://iclr.cc/virtual/2021/oral/3458

[71] Z. Liu et al., "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10012–10022.

[72] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7262–7272.

[73] X. Chu et al., "Twins: Revisiting the design of spatial attention in vision transformers," in *Proc. 35th Conf. Neural Inf. Process. Syst.*, 2021, pp. 9355–9366.

[74] E. Xie et al., "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 34, Dec. 2021, pp. 12077–12090.

[75] H. Zhang et al., "ResNeSt: Split-attention networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 2736–2746.

[76] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2020, pp. 173–190.

[77] K. He et al., "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, 2015.

[78] D. Tian, Y. Han, B. Wang, T. Guan, H. Gu, and W. Wei, "Review of object instance segmentation based on deep learning," *J. Electron. Imag.*, vol. 31, no. 4, Dec. 2021, Art. no. 041205.

[79] Z. Cai and N. Vasconcelos, "Cascade R-CNN: High quality object detection and instance segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1483–1498, May 2019.

[80] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring R-CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6409–6418.

[81] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-time instance segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9157–9166.

[82] Z. Tian, C. Shen, X. Wang, and H. Chen, "BoxInst: High-performance instance segmentation with box annotations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 5443–5452.

[83] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "SOLO: Segmenting objects by locations," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 649–665.

[84] X. Wang, R. Zhang, T. Kong, L. Li, and C. Shen, "SOLOv2: Dynamic and fast instance segmentation," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 17721–17732.

[85] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 1971–1980.

[86] F. Al-Turjman, H. Zahmatkesh, and R. Shahroze, "An overview of security and privacy in smart cities' IoT communications," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 3, p. e3677, Mar. 2022.

[87] L. Jiao et al., "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128837–128868, 2019.

[88] W. Gu, S. Bai, and L. Kong, "A review on 2D instance segmentation based on deep neural networks," *Image Vis. Comput.*, vol. 120, Apr. 2022, Art. no. 104401.

[89] Z. Tian, C. Shen, and H. Chen, "Conditional convolutions for instance segmentation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Aug. 2020, pp. 282–298.

[90] J. Dai et al., "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 764–773.

[91] S. Nienaber, R. S. Kroon, and M. J. Booysen, "A comparison of low-cost monocular vision techniques for pothole distance estimation," in *Proc. IEEE Symp. Ser. Comput. Intell.*, Dec. 2015, pp. 419–426.

[92] M. Everingham, L. van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–338, Jun. 2010.

[93] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.

Sicen Guo (Graduate Student Member, IEEE) is currently pursuing the Ph.D. degree with Tongji University. Her research interests include stereo matching and semantic segmentation.

Jiahang Li (Graduate Student Member, IEEE) is currently pursuing the M.Sc. degree with Tongji University. His research interests include computer vision and deep learning.

Yi Feng is currently pursuing the M.Sc. degree with Tongji University. His research interests include computer vision and deep learning.

Dacheng Zhou is currently pursuing the B.E. degree in automation with Tongji University.

Denghuang Zhang is currently pursuing the B.Sc. degree in automation with Tongji University.

Chen Chen is currently pursuing the B.E. degree in automation with Tongji University.

Shuai Su (Student Member, IEEE) is currently pursuing the Ph.D. degree with Tongji University. His research interests include computer vision and deep learning.

Xingyi Zhu is currently a Full Professor with the College of Transportation Engineering, Tongji University. Her research interests include intelligent functional pavement and the evaluation of aircraft landing resistance risk assessment.

Qijun Chen (Senior Member, IEEE) is currently a Full Professor with the College of Electronics and Information Engineering, Tongji University. His research interests include robotics control, environmental perception, and understanding of mobile robots and bioinspired control.

Rui Fan (Senior Member, IEEE) is currently a Full Professor with Tongji University and Shanghai Research Institute for Intelligent Autonomous Systems. His research interests include computer vision, deep learning, and robotics.