Supplementary Materials for "LEGNet: A Lightweight Edge-Gaussian Network for Low-Quality Remote Sensing Image Object Detection"

Wei Lu[1], Si-Bao Chen[1], Hui-Dong Li[1], Qing-Ling Shu[1], Chris H. Q. Ding[2], Jin Tang[1], Bin Luo[1]
[1]Anhui University, [2]The Chinese University of Hong Kong (Shenzhen)
{2858191255, 2563489133}@qq.com, {sbchen, tangjin, luobin}@ahu.edu.cn,
e23301341@stu.ahu.edu.cn, chrisding@cuhk.edu.cn

This supplementary document provides additional details to complement the main paper. It aims to facilitate reproducibility and deepen the understanding of our proposed LEGNet, including its architectural design, core components, and extensive experimental validation.

The appendix is organized as follows:

- **Appendix A** provides detailed descriptions of the datasets used for evaluation and the experimental setup.
- **Appendix B** presents the specific architectural configurations for the different variants of LEGNet (Tiny and Small).
- **Appendix C** includes an in-depth discussion on our key design choices, particularly the rationale for employing explicit priors and the sensitivity of hyperparameters.
- **Appendix D** offers a visual analysis of intermediate feature responses, providing insight into how LEGNet enhances feature representations compared to other methods.
- **Appendix E** showcases additional qualitative results, visualizing detection performance on the DOTA-v1.0 benchmark.

A. Datasets and Experimental Setup

As shown in Tab. 1, LEGNet was evaluated on five well-established datasets for object detection in remote sensing images: DOTA 1.0 [9], DOTA 1.5 [9], DIOR-R [2], FAIR1M-v1.0 [8], and VisDrone2019 [3]. These datasets provide diverse challenges, including variations in object scale, orientation, density, and environmental conditions, making them ideal for assessing the robustness of object detection models.

Dataset	Train Set	Test Set	Instances	Categories	Resolution	Key Features
DOTA-v1.0	21,046	10,833	188,282	15	1,024 × 1,024	High-resolution, diverse object sizes, aspect ratios, orientations
DOTA-v1.5	21,046	10,833	403,318	16	1,024 × 1,024	Adds small object annotations and container crane category, orientations
DIOR-R	11,725	11,738	192,472	20	800 × 800	high inter-class similarity, intra-class diversity, orientations
FAIR1M-v1.0	95,396	48,701	>1,000,000	5 (37 sub)	682 × 682, 1,024 × 1,024, 2,048 × 2,048	Large-scale, fine-grained categories, geographic metadata, orientations
VisDrone2019	6,471	548	>2,600,000	10	480 × 360 to 2,000 × 1,500 (Varies)	Drone-based, dense targets, complex backgrounds, diverse scenarios

Table 1. Comparison of Remote Sensing Object Detection Datasets

• DOTA-v1.0 and v1.5. The datasets are widely recognized benchmark for object detection in aerial images, featuring high-resolution images ranging from 800 × 800 to 20,000 × 20,000 pixels. The datasets were split into training (1,411 images), validation (458 images), and test sets (937 images). To meet the DOTA-v1.0 and v1.5 benchmark, images were divided into 1,024 × 1,024 patches with a 200-pixel overlap, resulting in approximately 21,046 patches for training and 10,833 for testing. Models were trained on the combined training and validation sets, and their performance was evaluated on the test set.

DOTA-v1.0 comprises 2,806 images with 188,282 annotated instances across 15 categories, including Plane (PL), Baseball diamond (BD), Bridge (BR), Ground track field (GTF), Small vehicle (SV), Large vehicle (LV), Ship (SH), Tennis court (TC), Basketball court (BC), Storage tank (ST), Soccer-ball field (SBF), Roundabout (RA), Harbor (HA), Swimming pool (SP), and Helicopter (HC). Its diversity in object sizes, orientations, and aspect ratios poses significant challenges for oriented object detection. DOTA-v1.5 builds upon DOTA-v1.0 by incorporating annotations for extremely small objects (less than 10 pixels) and introducing a new category, container crane (CC), increasing the total to 403,318 instances. This makes DOTA-v1.5 particularly suited for evaluating models on small and densely packed objects.

• **DIOR-R.** Derived from the DIOR dataset [5], DIOR-R is tailored for object detection in optical remote sensing images with oriented bounding box (OBB) annotations for precise localization. It includes 23,463 images, each at 800 $\times$ 800 pixels, with 192,472 annotated instances across 20 categories, such as airplanes, ships, and baseball fields. The dataset's high inter-class similarity and intra-class diversity challenge models to distinguish between visually similar objects.

• **FAIR1M-v1.0.** FAIR1M-v1.0 is one of the largest datasets for fine-grained object detection in high-resolution remote sensing images, containing 15,266 images with over 1 million annotated instances. These are organized into 5 main categories and 37 sub-categories, with images captured at resolutions of 0.3–0.8 meters. To ensure fairness, we follow the same dataset processing approach as LSKNet [6]. We adopt multi-scale training and testing strategy by first rescaling the images into three scales (0.5, 1.0, 1.5), and then cropping each scaled image into 1,024 $\times$ 1,024 sub-images with a patch overlap of 500 pixels, resulting in approximately 95,396 patches for training and 48,701 for testing. The dataset includes geographic metadata, such as latitude, longitude, and resolution, enhancing its utility for geospatial applications. Its scale and fine-grained categorization make it ideal for testing models on complex, large-scale detection tasks.

• **VisDrone2019.** The dataset is a comprehensive benchmark for drone-based object detection, consisting of 10,209 static images captured by various drone-mounted cameras across diverse urban and rural environments. It features over 2.6 million bounding box annotations across 10 categories, including pedestrians, cars, and bicycles, under varying weather and lighting conditions. The dataset's dense target distributions and complex backgrounds make it a challenging testbed for developing robust detection algorithms for unmanned aerial vehicle (UAV) images.

B. LEGNet Configuration Details

Stage	Downs. Rate	Layer Specification	(Input / Output) channels (C)	
			Tiny	Small
1	$\frac{H}{4} \times \frac{W}{4}$	`LoG-Stem Layer`	$3/32$	$3/64$
		`[LEG Block]` $\times N_1$	$32/32$	$64/64$
2	$\frac{H}{8} \times \frac{W}{8}$	`DRFD Module`	$32/64$	$64/128$
		`[LEG Block]` $\times N_2$	$64/64$	$128/128$
3	$\frac{H}{16} \times \frac{W}{16}$	`DRFD Module`	$64/128$	$128/256$
		`[LEG Block]` $\times N_3$	$128/128$	$256/256$
4	$\frac{H}{32} \times \frac{W}{32}$	`DRFD Module`	$128/256$	$256/512$
		`[LEG Block]` $\times N_4$	$256/256$	$512/512$
Number of Block			$[N_1, N_2, N_3, N_4] = [1, 4, 4, 2]$	

Table 2. Architecture configurations of LEGNet.

Tab. 2 provides the architectural configurations of LEGNet, which features two distinct scales (Tiny and Small) structured into four sequential stages. Each stage progressively downsamples the spatial resolution of the input feature maps. Each stage commences with an initial layer responsible for downsampling, followed by a series of repeated LEG Blocks ($N_i\times$).

Stage 1.

Downsampling Rate: The spatial resolution is reduced to $H/4 \times W/4$ relative to the original input dimensions.

Initial Layer: A 'LoG-Stem Layer' processes the input. For the Tiny configuration, it transforms 3 input channels to 32 output channels ($3/32$). For the Small configuration, it maps 3 input channels to 64 output channels ($3/64$). This indicates an initial feature extraction and channel expansion.

Core Blocks: This stage includes N_1 repetitions of the '[LEG Block]'. Both Tiny and Small versions maintain the channel dimensions within these blocks (Tiny: $32/32$; Small: $64/64$), indicating a focus on learning hierarchical features without

further channel changes at this sub-stage.
Stage 2.
Downsampling Rate: The spatial resolution is further reduced to $H/8 \times W/8$.
Initial Layer: A 'DRFD Module' is employed for inter-stage transition and downsampling. The Tiny model expands channels from 32 to 64 ($32/64$), while the Small model expands from 64 to 128 ($64/128$). This module incorporates downsampling operations to achieve the specified resolution reduction.
Core Blocks: This stage comprises N_2 '[LEG Block]' repetitions. Channels are maintained within these blocks (Tiny: $64/64$; Small: $128/128$), suggesting the primary role of these blocks is to refine features at the current resolution.
Stage 3.
Downsampling Rate: The spatial resolution is reduced to $H/16 \times W/16$.
Initial Layer: Another 'DRFD Module' facilitates the transition. The Tiny model maps channels from 64 to 128 ($64/128$), and the Small model maps from 128 to 256 ($128/256$). This continues the pattern of channel expansion and downsampling.
Core Blocks: N_3 '[LEG Block]' repetitions are used, preserving channel dimensions (Tiny: $128/128$; Small: $256/256$).
Stage 4.
Downsampling Rate: The final downsampling brings the resolution to $H/32 \times W/32$.
Initial Layer: The last 'DRFD Module' handles the transition. The Tiny model transforms channels from 128 to 256 ($128/256$), and the Small model from 256 to 512 ($256/512$).
Core Blocks: N_4 '[LEG Block]' repetitions are present, maintaining channel dimensions (Tiny: $256/256$; Small: $512/512$).

Block Repetitions. The number of 'LEG Block' repetitions for each stage, denoted as $[N_1, N_2, N_3, N_4]$, is consistently set to $[1, 4, 4, 2]$ for both Tiny and Small configurations.

C. Discussion on Design Choices

This section elaborates on key design choices within LEGNet, providing the rationale for embedding explicit priors over purely end-to-end learning and discussing the sensitivity of crucial hyperparameters.

C.1. Rationale for Employing Explicit Priors

A natural question regarding our approach is why we chose to explicitly encode priors, such as edge and Gaussian features, rather than relying on a sufficiently deep or complex network (e.g., a Transformer-based model) to learn them automatically. While modern deep networks possess immense learning capabilities, our approach of embedding explicit priors offers several distinct advantages, particularly in the context of our goal to build a lightweight and robust backbone for RSOD.

- **Data and Parameter Efficiency:** Learning fundamental concepts like edges or Gaussian-like attention from scratch is a parameter-intensive task. By embedding these priors through parameter-free operators (our LoG-Stem and the fixed Scharr/Gaussian filters in the EGA module), we provide the network with a strong, built-in "head start." This makes the model significantly more data-efficient and allows it to achieve high performance with a much lower parameter count—a core objective of our lightweight LEGNet design.
- **Robustness to Degradation:** This is a cornerstone of our motivation. End-to-end models learn features based on statistical patterns in the training data. When input images are degraded (e.g., due to blur, low contrast, or noise), these patterns can become weak or distorted, causing learned filters to fail. In contrast, our explicit edge detectors are deterministic operators that can reliably extract structural information even from degraded signals. By providing the network with this robust edge map, we ensure that subsequent layers receive meaningful structural cues, enhancing the model's resilience to poor imaging conditions.
- **Guided Learning and Regularization:** Explicit priors act as a strong inductive bias, guiding the network to focus on structurally relevant information from the earliest stages. This serves as a form of regularization, discouraging the model from overfitting to spurious textures or background noise. As discussed in our macro design (**??**), this biases the model towards learning "the right features for the right reasons," leading to better generalization.
- **Interpretability:** The use of well-defined operators like Laplacian of Gaussian provides a degree of interpretability to the early-stage feature extraction process. We know precisely that the initial layers are enhancing edges, which aligns with human visual processing and provides clearer insight into the model's behavior, as supported by our feature visualizations in Fig. 1 of the Appendix.

In summary, our choice is not based on the premise that deep networks *cannot* learn these features, but rather that explicitly encoding them provides a more efficient, robust, and targeted pathway to building a high-performing lightweight model for the specific challenges of RSOD.

C.2. Discussion on Hyperparameter Sensitivity

The hyperparameters of the Gaussian kernel, namely its size and bandwidth (standard deviation, σ), are influential components of the EGA module. While a full ablation study was beyond the scope of our primary investigation, we discuss their roles and the reasoning behind our choices here.

- **Gaussian Kernel Size:** The kernel size determines the spatial extent of the feature aggregation. A larger kernel incorporates context from a wider neighborhood, beneficial for larger objects, but risks over-smoothing details and blurring together small, dense objects. Conversely, a smaller kernel preserves fine details but may fail to capture sufficient context. Our selected kernel size was determined during preliminary experiments to strike a balance suitable for the multi-scale nature of objects in datasets like DOTA and FAIR1M. The chosen value proved effective at capturing salient features without significant information loss.
- **Gaussian Bandwidth (σ):** The bandwidth controls the decay rate of the Gaussian function. A small σ creates a sharp filter that heavily prioritizes the central feature, while a large σ creates a smoother filter that gives more uniform weight to the neighborhood. The value of σ is often set proportionally to the kernel size to ensure the distribution fits naturally within the kernel's window. Our implementation follows this standard practice, a well-established heuristic in computer vision for ensuring stability.

Crucially, while these parameters are important, we found the model's performance to be robust to minor variations around our chosen values. This robustness is largely because the subsequent learnable layers (e.g., the 1x1 convolutions within our LEG Blocks) grant the network the flexibility to adapt and recalibrate the features generated by the fixed-parameter Gaussian module. Therefore, the overall architecture is not hyper-sensitive to these specific values, and the selected parameters represent a stable and effective configuration for the task.

D. Analysis of Intermediate Feature Responses

To provide deeper insight into LEGNet's mechanism, we visualize the intermediate feature maps. As shown in Fig. 1, we compare the feature responses of our LEGNet-S with a strong competitor, PKINet-S, at different stages of the backbone: the initial stem layer, the output of Stage 1, and the output of Stage 2.

The visualizations reveal distinct differences in feature representation. In the stem layer and Stage 1, LEGNet (bottom row) demonstrates a superior ability to extract comprehensive and complete edge information across the entire image compared to PKINet (top row). This is attributed to our LoG-Stem layer and the EGA module, which explicitly enhance structural details from the outset. As the network deepens into Stage 2, LEGNet exhibits a more focused attention on salient object features, effectively suppressing background noise and highlighting regions of interest. In contrast, the feature responses from PKINet appear more diffuse. This comparison visually substantiates our claim that by integrating explicit edge and Gaussian priors, LEGNet learns more robust and meaningful feature representations, particularly in the crucial early stages of processing. Due to the highly abstract nature of features in deeper stages (3 and 4), we focus on these initial stages where the impact of our design is most visually interpretable.

E. Visualization Results

The visual results on the DOTA-v1.0 test set are presented in Figs. 2 to 6. We conducted a visual analysis using SOTA backbones, including ARC-R50 [7] and PKINet [1], both of which were designed specifically for RSOD tasks. The visualization results of ARC-R50 were based on the MMDetection toolbox, while ResNet-50 [4], PKINet-S and LEGNet-S visualization results were based on the MMRotate toolbox. All models were developed based on the O-RCNN [10] detector.

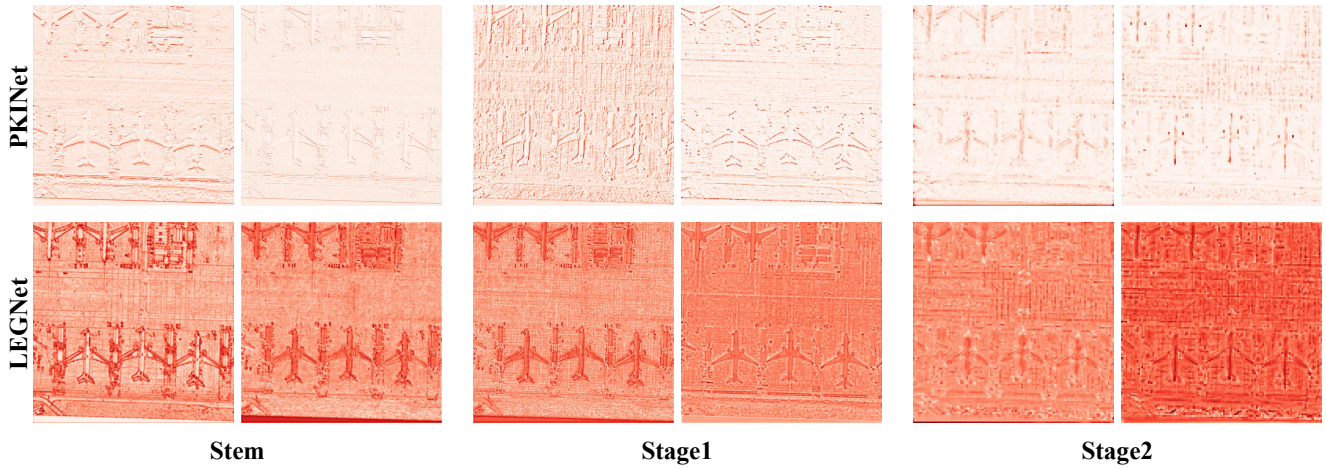

Figure 1. Visualization of intermediate feature maps from PKINet-S (top row) and our LEGNet-S (bottom row). LEGNet's early stages capture more complete edge details, while Stage 2 demonstrates enhanced focus on salient object regions. This supports the effectiveness of our proposed design in refining feature representations for robust detection.

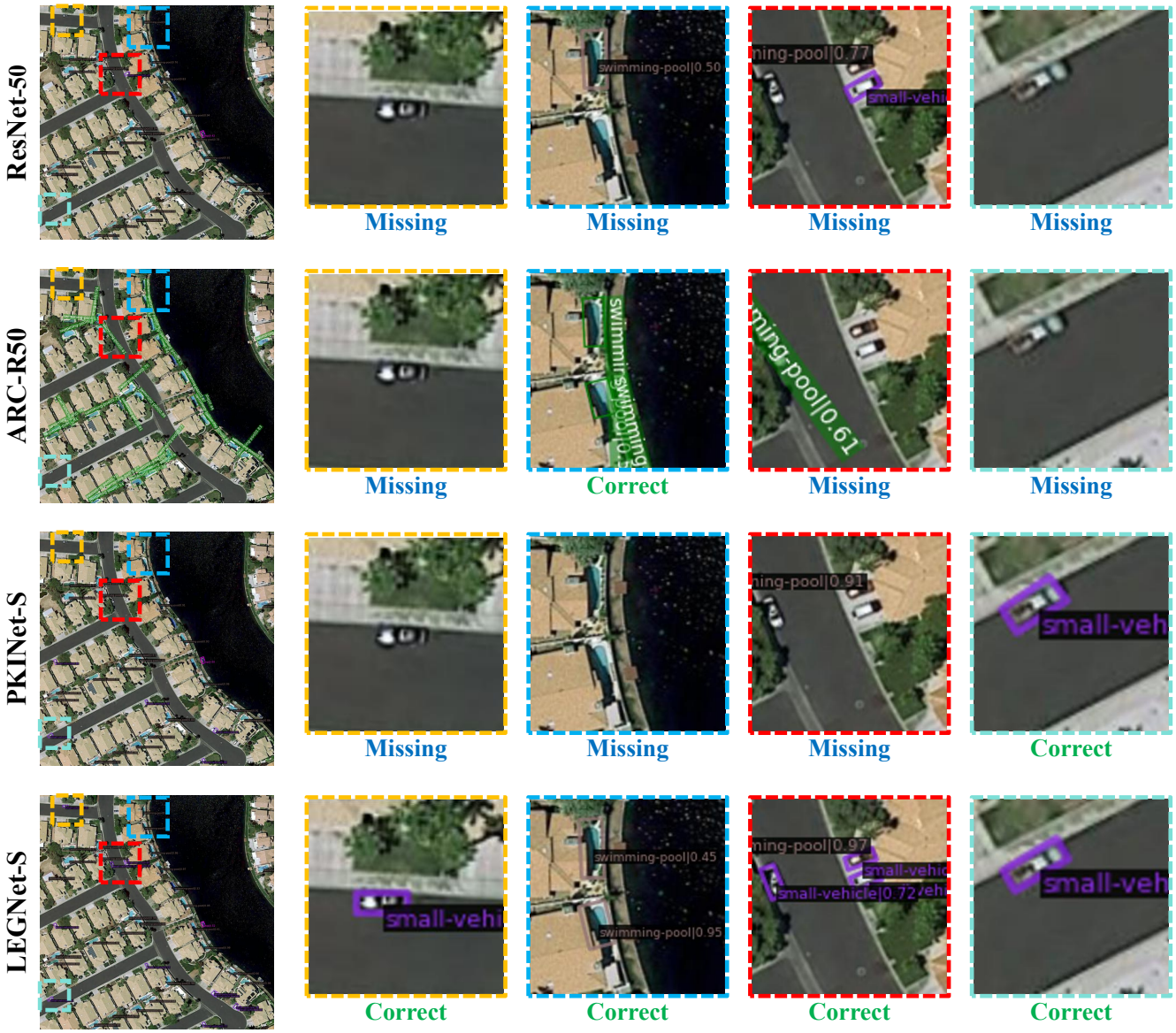

Figure 2. Visualization of detection results on DOTA-v1.0 test set. Input images resolution were 1,024 × 1,024.

Figure 3. Visualization of detection results on DOTA-v1.0 test set. Input images resolution were 1,024 × 1,024.

Figure 4. Visualization of detection results on DOTA-v1.0 test set. Input images resolution were 1,024 × 1,024.

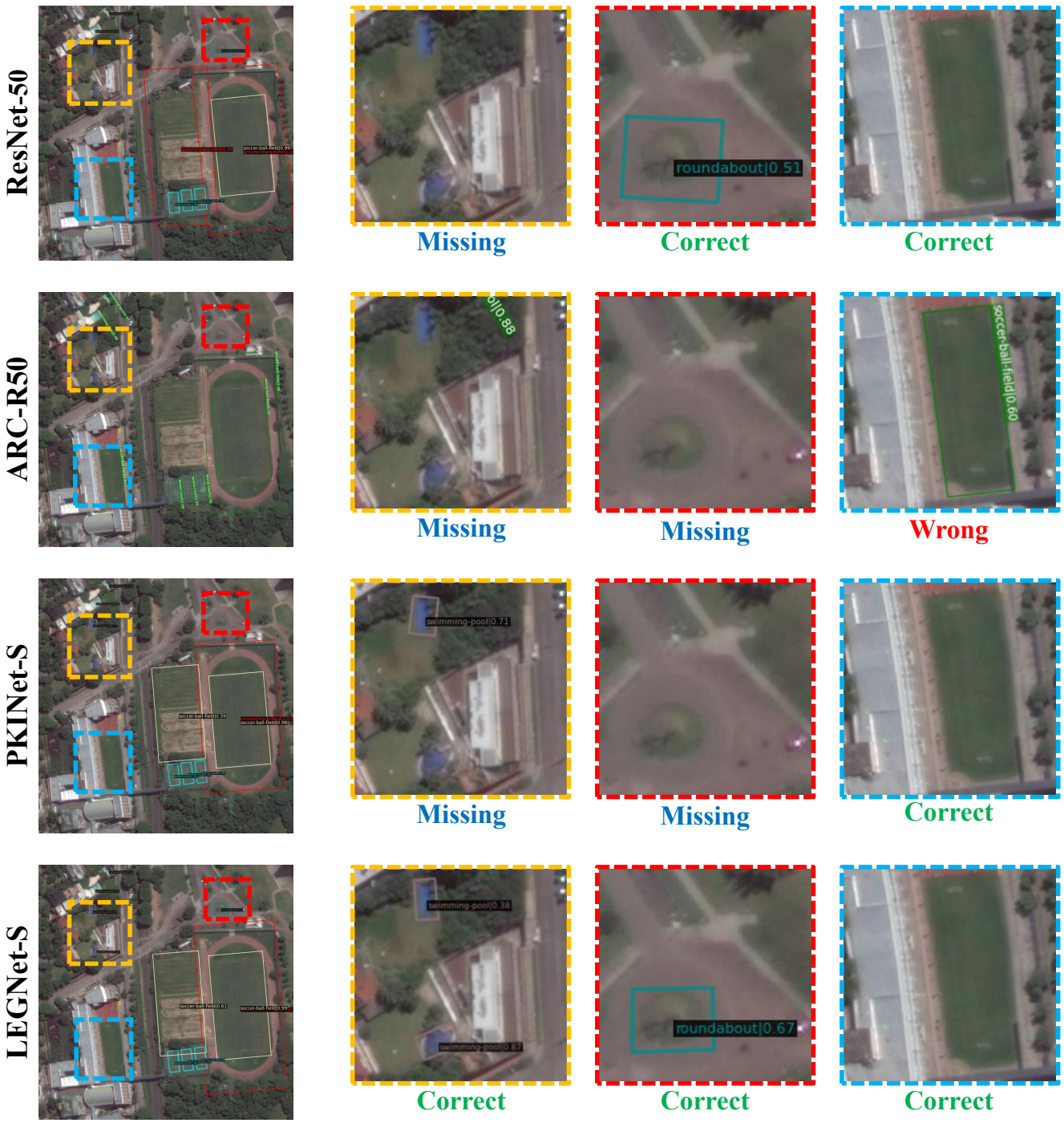

Figure 5. Visualization of detection results on DOTA-v1.0 test set. Input images resolution were 1,024 × 1,024.

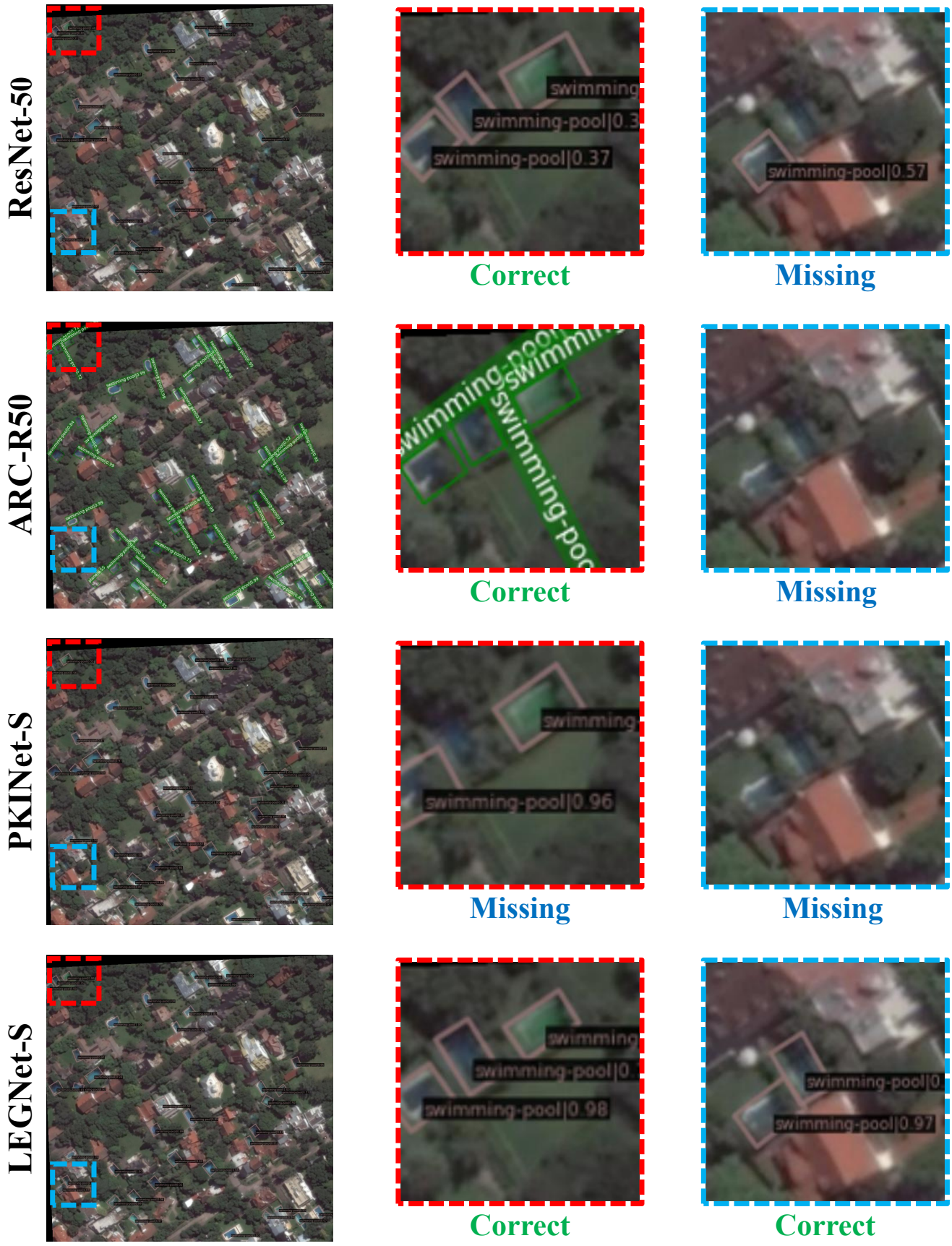

Figure 6. Visualization of detection results on DOTA-v1.0 test set. Input images resolution were 1,024 × 1,024.

References

[1] Xinhao Cai, Qiuxia Lai, Yuwei Wang, Wenguan Wang, Zeren Sun, and Yazhou Yao. Poly kernel inception network for remote sensing detection. In *CVPR*, pages 27706–27716, 2024. 4

[2] Gong Cheng, Jiabao Wang, Ke Li, Xingxing Xie, Chunbo Lang, Yanqing Yao, and Junwei Han. Anchor-free oriented proposal generator for object detection. *IEEE Trans. Geosci. Remote Sens.*, 60:1–11, 2022. 1

[3] Dawei Du, Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Lin, Qinghua Hu, Tao Peng, Jiayu Zheng, Xinyao Wang, Yue Zhang, et al. Visdrone-det2019: The vision meets drone object detection in image challenge results. In *ICCVW*, 2019. 1

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4

[5] Ke Li, Gang Wan, Gong Cheng, Liqiu Meng, and Junwei Han. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS J. Photogramm. Remote Sens.*, 159:296–307, 2020. 2

[6] Yuxuan Li, Qibin Hou, Zhaohui Zheng, Ming-Ming Cheng, Jian Yang, and Xiang Li. Large selective kernel network for remote sensing object detection. In *ICCV*, pages 16794–16805, 2023. 2

[7] Yifan Pu, Yiru Wang, Zhuofan Xia, Yizeng Han, Yulin Wang, Weihao Gan, Zidong Wang, Shiji Song, and Gao Huang. Adaptive rotated convolution for rotated object detection. In *ICCV*, pages 6589–6600, 2023. 4

[8] Xian Sun, Peijin Wang, Zhiyuan Yan, Feng Xu, Ruiping Wang, Wenhui Diao, Jin Chen, Jihao Li, Yingchao Feng, Tao Xu, et al. FAIR1M: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery. *ISPRS J. Photogramm. Remote Sens.*, 184:116–130, 2022. 1

[9] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *CVPR*, pages 3974–3983, 2018. 1

[10] Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. Oriented r-cnn for object detection. In *ICCV*, pages 3520–3529, 2021. 4