remote sensing

MDPI

Article

Scale-Aware Gated Routing Distillation for Lightweight Remote Sensing Object Counting

Zhiyi Zhou [1,†], Junjing Zheng [1,†], Yihang Gao [2] and Hongying Meng [3,*]

[1] School of International Studies, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; 2378903@brunel.ac.uk (Z.Z.); 2378933@brunel.ac.uk (J.Z.)
[2] School of Automation, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; 2024213100@stu.cqupt.edu.cn
[3] Department of Electronic and Electrical Engineering, College of Engineering, Design and Physical Sciences, Brunel University of London, Uxbridge UB8 3PH, UK
* Correspondence: hongying.meng@brunel.ac.uk
† These authors contributed equally to this work.

Highlights

What are the main findings?

- SAGRD builds a scale-aware gated routing pyramid that adaptively reweights dilated receptive-field branches using spatial logits, branch feedback, and variance-based scale descriptors.
- SAGRD distills not only density maps but also teacher routing distributions, helping the lightweight student inherit scale-adaptive counting behavior.

What are the implications of the main findings?

- SAGRD-Student achieves 4.712 MAE and 6.367 RMSE on RSOC Building with only 1.395 M parameters and 1.29 G FLOPs.
- Compared with the teacher, the student reduces parameters and FLOPs by about 90.2% and 91.3%, while preserving interpretable routing responses for deployment-friendly remote sensing counting.

Abstract

Remote sensing object counting remains challenging for lightweight models due to large scale variation, dense object distribution, and complex background interference. Existing lightweight counting methods usually rely on static multi-scale fusion, which limits adaptive receptive-field selection under heterogeneous scenes. To address this issue, this paper proposes Scale-Aware Gated Routing Distillation (SAGRD), a lightweight framework that reformulates multi-scale feature fusion as an adaptive routing problem. At its core, a scale-aware gated routing pyramid generates dynamic branch weights from three complementary cues—global scale descriptors, branch-response feedback, and spatial routing signals—to adaptively reweight dilated receptive-field responses before weighted aggregation, enabling more flexible scale-aware representation under limited model capacity. In addition, a routing-oriented distillation strategy is introduced to transfer not only density estimation knowledge but also the teacher's scale-selection distributions to the lightweight student, allowing the compact model to inherit structured receptive-field adaptation strategies. Extensive experiments on multiple remote sensing object counting datasets demonstrate that the proposed framework achieves a favorable accuracy-efficiency trade-off. On the RSOC Building benchmark, the student model achieves an MAE/RMSE of 4.712/6.367 using only 1.395 M parameters and 1.29 G FLOPs, closely approaching the teacher performance.

Check for updates

Academic Editor: Salah Bourennane

Received: 4 July 2026
Revised: 20 August 2026
Accepted: 26 August 2026
Published: 1 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Keywords: remote sensing object counting; lightweight neural network; density map estimation; knowledge distillation; scale-aware routing; dynamic gating

1. Introduction

Remote sensing dense object counting is challenging because it requires high accuracy under strict efficiency constraints. Overhead images often contain objects that vary substantially in size, density, and spatial layout. In such scenes, a successful model must preserve fine-grained local responses for small and dense targets while retaining sufficient contextual information for larger or spatially dispersed objects. This requirement becomes even more difficult for lightweight models, because reduced capacity makes it harder to adapt to complex scenes. In practice, the key challenge is therefore not simply to make the model smaller, but to discover a mechanism which can effectively focus on both local and global context information [1,2].

Existing counting methods usually improve performance by introducing multi-column structures, pyramids, contextual aggregation, or lightweight feature fusion. However, most of these designs still rely on static fusion rules, such as concatenation, summation, or globally fixed weights. Such mechanisms indeed can improve representation ability and capacity, but they do not explicitly tell which size of receptive-field should be emphasized for a specific region or scene. For lightweight counting models, this limitation is quite obvious: it either over-compresses scale information or increases complexity while trying to cover all possible object sizes [3–5].

Conditional feature weighting provides an alternative to static multi-scale fusion through selective kernels [6], dynamic convolution [7], conditional parameterization [8], and mixture-of-experts (MoE) routing [9,10]. Recent studies have further extended MoE-style routing to dense prediction, including multi-task dense prediction [11] and CNN-based semantic segmentation [12]. These studies show that routing behavior and expert specialization are sensitive to architectural and optimization choices. In lightweight counting, this motivates explicit stabilization of router–expert co-optimization. Existing counting distillation methods also mainly transfer density outputs or intermediate features, leaving the teacher's routing preferences untransferred. SAGRD therefore focuses on making scale-aware soft routing both stable and transferable to a compact student.

To address this gap, this paper proposes Scale-Aware Gated Routing Distillation (SAGRD). SAGRD treats multi-scale feature fusion as a soft routing problem. The scale-aware gated routing pyramid constructs several receptive-field branches and assigns them content-dependent weights using spatial structure, branch-response feedback, and global scale descriptors. All branches are evaluated, while their contributions are adaptively reweighted before aggregation. SAGRD further introduces routing-oriented distillation, through which the student learns both density predictions and branch-weight distributions from the teacher, thereby inheriting counting behavior and structured receptive-field preferences under limited capacity.

The main contributions of this paper are as follows:

- We propose SAGRD, a lightweight remote sensing object counting framework that formulates multi-scale feature fusion as soft adaptive routing. The framework assigns content-dependent weights to receptive-field branches rather than using fixed aggregation.
- We design a stabilization strategy for lightweight dynamic routing. It includes gradient-decoupled branch feedback, temperature annealing, and complexity-conditioned gate diversity regularization. These components reduce premature branch domination and make expert specialization more reliable.

- We introduce routing-oriented knowledge distillation. The student receives density estimation guidance and scale-selection guidance from the teacher. Image-level routing alignment helps the compact model learn structured receptive-field preferences.
- Experiments on multiple remote sensing object counting datasets show a favorable accuracy–efficiency trade-off. On RSOC Building, the student model achieves 4.712 MAE and 6.367 RMSE with only 1.395 M parameters and 1.29 G FLOPs. It remains close to the teacher while outperforming several lightweight counting baselines.

2. Related Work

2.1. Remote Sensing Object Counting

Remote sensing object counting aims to estimate the number of target instances in overhead imagery and supports applications such as traffic monitoring, urban analysis, port management, and infrastructure assessment. Compared with natural-scene counting, remote sensing images usually contain smaller targets, larger scale variation, and more complex background interference. These characteristics make dense object counting in remote sensing scenarios particularly challenging.

To handle high-density object distributions, density-map-based counting has become a widely adopted paradigm [13]. Instead of detecting each object individually, density-map methods convert object annotations into continuous spatial density distributions and estimate the final count by integrating the predicted density map. Formally, the Dirac delta function is adopted to represent the point-level counting label [4], which satisfies $\delta(x) = 0$ for $x \neq 0$ and $\int_{-\infty}^{+\infty} \delta(x)dx = 1$. Let $\{x_i\}_{i=1}^{N}$ denote the set of annotated object centers on the image plane. These discrete labels can be aggregated as a sum of Dirac impulses $H(x) = \sum_{i=1}^{N} \delta(x - x_i)$, which is then convolved with an isotropic Gaussian kernel G_σ of standard deviation σ to produce the continuous ground-truth density map:

$$D(x) = H(x) * G_\sigma(x) = \sum_{i=1}^{N} \delta(x - x_i) * G_\sigma(x), \quad C = \sum_{x} D(x). \tag{1}$$

where $*$ denotes the convolution operation, x_i represents the position of the i-th annotated object center, and $G_\sigma(\cdot)$ denotes the Gaussian kernel function [13].

Early remote sensing counting methods were largely adapted from natural-scene crowd counting, including multi-column density regression, dilated convolution, and context-aware aggregation networks [3,14,15]. However, the introduction of comprehensive benchmarks, ranging from satellite imagery in RSOC, which covers buildings, ships, and large/small vehicles, to drone- and surveillance-view vehicle datasets such as CARPK and PUCPR+, revealed domain-specific challenges in remote sensing and overhead object counting [1,2,16]. These datasets show that extreme scale variation, viewpoint changes, dense spatial distributions, and background clutter can severely degrade counting accuracy. Recent studies in *remote sensing* have reached similar conclusions: SwinCounter introduced transformer-based multi-scale information fusion and a balanced loss for RSOC, while TASNet combined triple-view attention with scale-aware feature aggregation to address background clutter and scale variation [17,18]. To address these issues, subsequent methods have explored scale-aware density estimation, context aggregation, deformable convolutions, pyramidal feature representation, and multi-branch feature extraction [2,19,20]. While improving robustness across complex scenes, these advances often introduce substantial computational overhead, motivating more efficient and lightweight adaptation strategies for deployment-oriented remote sensing object counting [21,22].

2.2. Multi-Scale Receptive-Field Modeling

Multi-scale receptive-field modeling is a central issue in remote sensing object counting, because object size, density, and spatial distribution may vary markedly across scenes and even within the same image. Small vehicles and densely distributed buildings usually require fine local responses to preserve spatial details, whereas ships, large vehicles, and sparse building regions often benefit from broader contextual perception to suppress background interference. Effective counting models therefore need to balance local discrimination and contextual aggregation under different scale distributions.

Early density-map-based counting methods addressed scale variation through multi-column architectures. MCNN [14] is a representative example, where parallel convolutional branches with different kernel sizes are used to capture targets at different scales. This design explicitly introduces receptive-field diversity and has inspired many subsequent counting networks. However, multi-column structures often increase computational cost and usually combine branch outputs through simple concatenation or summation, making the scale-fusion process relatively static.

Another widely used strategy is dilated convolution, which enlarges the receptive field without reducing feature-map resolution. CSRNet [15] adopted dilated convolution to aggregate broader contextual information while maintaining dense spatial prediction ability. This design is effective for high-density scenes because it avoids excessive downsampling and preserves the resolution required for density-map regression. Nevertheless, the dilation pattern is fixed after training in most cases, and thus it may not fully adapt to input-dependent scale variations in remote sensing imagery.

Beyond fixed receptive-field expansion, several methods introduce context-aware modeling to improve robustness under complex scenes. CAN [3] incorporates contextual information from multiple spatial ranges to enhance density estimation in cluttered environments. Such context aggregation helps distinguish targets from background patterns and improves global consistency of the predicted density map. However, context modules typically emphasize feature enrichment rather than explicit receptive-field selection, and their responses are still constrained by predefined aggregation structures.

In remote sensing object counting, scale modeling has been further explored through pyramidal and scale-guided structures. The RSOC benchmark [2] highlighted the importance of scale-aware density estimation for buildings, vehicles, and ships in overhead imagery. ASPDNet [2] introduced scale-pyramid representation and deformable convolution to improve the modeling of objects with different sizes and orientations. PSGCNet [19] further combined pyramidal scale information with global context guidance, allowing the network to better handle dense object distributions in remote sensing scenes. In parallel, SwinCounter exploited transformer attention to capture multi-scale information in remote sensing object counting, and TASNet used a scale-aware branch design to strengthen multi-scale feature fusion under background clutter [17,18]. These methods demonstrate that scale-aware representation is essential for remote sensing counting, especially under large spatial-resolution variation and cluttered backgrounds.

Other studies have investigated multi-scale context aggregation, attention mechanisms, and feature-fusion strategies. For example, distillation-based remote sensing counting methods use multi-scale context feature aggregation to improve the learning of compact models [23]. Regression ensemble and learning-to-rank strategies have also been introduced to enhance robustness across different density ranges [24]. Lightweight multi-scale fusion networks, such as LMSFFNet [21], attempt to reduce computational cost while preserving useful scale information. Attention-based counting models, such as SFANet [25], further exploit spatial or channel-wise weighting to emphasize informative regions and suppress irrelevant responses.

Although these methods improve scale robustness from different perspectives, most of them still rely on relatively static multi-scale fusion. Multi-branch features are commonly concatenated, summed, or fused using fixed or globally shared weights. Such mechanisms increase representation diversity, but they do not explicitly determine which receptive-field branch should be emphasized for a particular scene or spatial region. This limitation is especially important in remote sensing imagery, where dense parking lots, sparse roads, port areas, and building clusters may require different receptive-field preferences. In contrast, the proposed SAGRD treats multi-scale fusion as an adaptive routing problem and learns content-dependent branch selection, providing a more flexible form of scale-aware receptive-field modeling.

2.3. Conditional Computation

Beyond static multi-scale fusion, conditional computation has been explored as a way to make feature processing depend on the input content. Selective Kernel Networks (SKNet) [6] introduce a dynamic selection mechanism that allows each neuron to adapt its effective receptive field by softly combining multiple branches with different kernel sizes. Dynamic convolution [7] further generalizes this idea by aggregating several convolution kernels with input-dependent attention, increasing representational capacity without enlarging network depth or width. CondConv [8] follows a related principle but parameterizes convolution conditionally for each input example, enabling the network to learn specialized kernels while retaining efficient inference. Together, these methods suggest that the most appropriate receptive field or kernel configuration is often content dependent rather than globally fixed.

A closely related formulation is mixture-of-experts (MoE), where a router assigns input-dependent weights or sparse assignments to multiple experts [9,10]. MoE-style designs have also been extended to dense prediction. MLoRE uses low-rank experts for multi-task dense prediction [11], while Pavlitska et al. [12] analyze sparse convolutional experts for semantic segmentation and report strong sensitivity of routing behavior and expert specialization to design choices. These results support the use of adaptive expert weighting beyond classification, while also motivating explicit routing stabilization. SAGRD follows this direction but uses a small soft-routed receptive-field expert set for remote sensing counting and evaluates the need for stabilization directly through the ablations in Section 4.5.

2.4. Lightweight Counting Networks

Lightweight counting networks seek to reduce computational cost while retaining reliable density-map estimation. This requirement is particularly important in remote sensing, where counting models may be deployed on embedded platforms, unmanned aerial vehicles, and real-time monitoring systems. Unlike classification tasks, object counting requires both precise spatial localization and contextual dependency modeling. Excessive compression may therefore degrade density-map quality, especially in dense and complex scenes.

General lightweight backbones have provided useful building blocks for efficient counting models. MobileNet [26] introduced depthwise separable convolution, MobileNetV2 [27] improved feature reuse with inverted residual blocks, and ShuffleNetV2 [28] enhanced channel interaction through efficient channel shuffle operations. MobileViT [29] further combined lightweight convolution with transformer-style global modeling, offering a compact way to capture local and long-range dependencies. These architectures have been widely adopted for efficient dense prediction.

In object counting, MobileCount [30] adapted MobileNetV2 into an encoder–decoder framework for real-time crowd counting. Lw-Count [31] and LEDCrowdNet [32] further designed lightweight encoding–decoding structures to reduce complexity while maintaining spatial reconstruction ability. For remote sensing object counting, LMSFFNet [21] proposed a lightweight multi-scale fusion network to balance scale robustness and efficiency, while EdgeCount [22] introduced density-map knowledge distillation to train a compact student model from a stronger teacher.

Although these methods significantly reduce parameters and FLOPs, most lightweight counting models still face a limitation: they mainly rely on backbone compression, channel reduction, or static feature fusion. Such strategies may weaken scale representation and are less adaptive to remote sensing scenes with different density levels. Therefore, preserving adaptive receptive-field selection under limited model capacity remains an open challenge, which motivates the design of SAGRD.

2.5. Knowledge Distillation for Counting

Knowledge distillation has become an effective strategy for constructing lightweight object counting models [33]. Its basic idea is to transfer knowledge from a high-capacity teacher network to a compact student network, so that the student can retain part of the teacher's representation ability under a much smaller computational budget. Compared with direct capacity compression, distillation often provides more stable optimization and better performance.

In remote sensing object counting, distillation is particularly useful because density maps contain rich spatial supervision beyond image-level counting labels. A common distillation formulation minimizes the discrepancy between teacher and student predictions:

$$\mathcal{L}_{\text{out}} = \|\hat{D}_s - \hat{D}_t\|_2^2, \tag{2}$$

where $\hat{D}_s$ and $\hat{D}_t$ denote the student and teacher density maps, respectively. Existing methods have further explored feature-level distillation, attention transfer, and intermediate representation alignment to improve spatial consistency between teacher and student networks.

Most distillation methods for counting focus on transferring density predictions, intermediate feature representations, or attention responses from the teacher network to the student network. Density-level distillation improves output consistency, while feature-based distillation attempts to preserve richer spatial representation ability in compact models. Some studies further introduce relational or attention-guided distillation to enhance structural similarity between teacher and student feature spaces.

In recent years, distillation methods have gradually moved from output supervision toward more structured representation transfer. Instead of only matching final predictions, several approaches also consider transferring internal response distributions, attention maps, or dynamic feature interactions [34]. These developments suggest that effective distillation for dense prediction tasks may benefit not only from prediction alignment, but also from preserving intermediate structural information within the teacher network. However, existing counting distillation methods rarely transfer the teacher's scale-selection behavior explicitly. This limitation motivates the routing-oriented distillation strategy adopted in SAGRD, where density estimation knowledge and receptive-field routing preferences are jointly distilled into the lightweight student.

3. Methods

3.1. Overview of the Proposed Framework

The overall architecture of the Scale-Aware Gated Routing Distillation (SAGRD) framework is illustrated in Figure 1. SAGRD is designed for lightweight remote sensing object counting under complex scale variation and dense spatial distribution.

Unlike conventional lightweight counting frameworks that mainly rely on static multi-scale fusion or output-level distillation, SAGRD reformulates multi-scale feature aggregation as a soft routing problem. The central motivation is to preserve input-dependent receptive-field adaptation despite the limited representation capacity of a lightweight model. SAGRD therefore evaluates a small set of parallel receptive-field branches and jointly infers their content-dependent weights from global scale descriptors, branch-response feedback, and spatial routing cues. The routing mechanism changes the contribution of each branch rather than skipping branch computation. Since dynamic routing in compact models is prone to branch collapse and unstable co-training, SAGRD further integrates a stabilization suite comprising gradient-decoupled feedback, temperature annealing, and gate diversity regularization. In addition, routing-oriented knowledge distillation transfers both density estimation knowledge and the teacher's scale-selection distributions to the lightweight student, ensuring that the compact model inherits structured receptive-field adaptation strategies rather than merely approximating counting outputs. As illustrated in Figure 1, the overall framework comprises three hierarchical stages—detail, semantic, and context—followed by cross-scale fusion and a lightweight density estimation head.

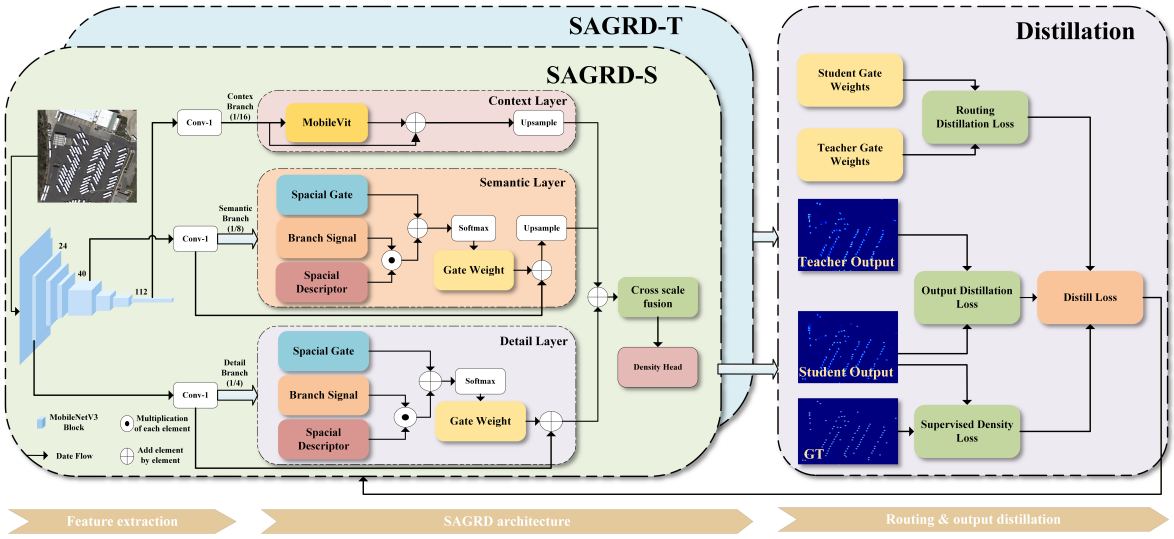


Figure 1. Overall architecture of the proposed SAGRD framework. SAGRD-T extracts stronger multi-scale features and produces both density-map predictions and gated routing distributions. Then, SAGRD-S learns from ground-truth density maps, teacher density maps, and teacher gated branch-weight distributions.

3.2. Gated Routing Signal

The core of SAGRD is to predict input-dependent branch weights for adaptive receptive-field aggregation. To this end, SAGRD first constructs multi-scale expert responses as candidate receptive-field routes, and then derives routing logits from three complementary cues: expert-response feedback, a global scale descriptor, and spatial routing information. The expert responses provide branch-specific activation evidence, the global descriptor offers an image-level scale prior, and the spatial signal supplies local structural cues. These cues are jointly used to produce dynamic branch weights for

scale-aware feature routing. Additionally, SAGRD is implemented under a teacher–student paradigm, where the teacher provides stronger scale-selection supervision to guide the lightweight student. The detailed teacher and student configurations are summarized in Table 1.

Table 1. Detailed teacher and student model configurations.

Component	Teacher	Student
Backbone	ResNet50 [35]	MobileNetV3-Large [36]
Detail/Semantic/Context channels	96/128/128	48/64/64
Normalization	Group Normalization [37]	Batch Normalization
Branch dilation rates	[2, 4, 6, 8]	[2, 4, 6, 8]
Expert block type	Standard dilated conv	Depthwise separable conv
Detail hidden dim.	96	24
Semantic hidden dim.	128	32
Spatial routing branch	DCNv2 + Standard conv	DCNv2 + Depthwise separable conv
Transformer dimension	128	32
Attention heads	8	4
Transformer layers	3	2
MLP ratio	4.0	2.0
Patch size/Dropout	2/0.2	2/0.2
Fusion hidden/output channels	192/160	96/96
Fusion module	SCConv [38] + Residual	SCConv [38] + Residual
Density head channels	160	96

3.2.1. Multi-Scale Expert Responses

Remote-sensing counting scenes often contain targets with substantially different sizes and spatial densities, making a single fixed receptive field insufficient for reliable density estimation. SAGRD therefore constructs K parallel expert branches $\{\mathcal{E}_k\}_{k=1}^{K}$ as candidate receptive-field routes for subsequent gated aggregation. Given an input feature map $\mathbf{F} \in \mathbb{R}^{B\times C\times H\times W}$, the k-th expert produces a branch-specific feature map

$$\mathbf{B}_k = \mathcal{E}_k(\mathbf{F}) \in \mathbb{R}^{B\times C'\times H\times W}, \quad k = 1, \ldots, K, \tag{3}$$

where each $\mathcal{E}_k(\cdot)$ denotes a dilated residual block with output channels C' and dilation rate $d_k \in \{2, 4, 6, 8\}$. This configuration provides dense coverage of moderate receptive-field scales: $d = 1$ is partly redundant with local backbone responses, whereas very large dilation such as $d = 16$ samples a wider but sparser context that can weaken fine spatial evidence. Each $\mathbf{B}_k$ retains the same spatial resolution $H \times W$ while encoding a different contextual range. We set $K = 4$, and the dilation-rate sensitivity is evaluated in Section 4.5. Rather than statically concatenating or averaging the multi-scale features, the branch responses are leveraged as dynamic feedback for routing prediction. Specifically, the branch outputs are concatenated along the channel dimension and projected to a compact response tensor:

$$\mathbf{R} = \mathcal{W}_r(\mathrm{Concat}_C[\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_K]), \quad \mathbf{R} \in \mathbb{R}^{B\times K\times H\times W}, \tag{4}$$

where $\mathrm{Concat}_C[\cdot]$ denotes channel-wise concatenation and $\mathcal{W}_r$ is a 1×1 convolutional layer that projects the aggregated features onto K channels.

To support routing-oriented distillation under lightweight constraints, the teacher and student adopt a routing-topology-aligned but capacity-asymmetric expert design. Both models share the same branch number K and the identical dilation configuration $\{d_k\}_{k=1}^{K}$, ensuring that their branch-weight distributions remain comparable for gate distillation. However, the teacher employs stronger standard dilated residual experts, whereas the student substitutes depthwise-separable residual experts to reduce computational cost.

Hence, the expert-response feedback is topologically aligned between teacher and student, yet its representation capacity is intentionally asymmetric.

3.2.2. Global Scale Descriptor

To provide an image-level scale prior for gated receptive-field routing, SAGRD employs a multi-resolution global scale descriptor. Since objects in remote sensing images exhibit substantial scale variation, an image-level summary of spatial heterogeneity helps the router establish coarse-scale tendencies before making local routing decisions. Given an input feature map $\mathbf{F} \in \mathbb{R}^{B\times C\times H\times W}$, adaptive average pooling layers are first applied to produce fixed-resolution pyramid representations:

$$\mathbf{P}_r = \mathrm{AAP}_r(\mathbf{F}) \in \mathbb{R}^{B\times C\times r\times r}, \quad r \in \{1,3,7\}, \tag{5}$$

where $\mathrm{AAP}_r(\cdot)$ denotes adaptive average pooling that spatially resizes feature map to a fixed resolution of $r \times r$. The three resolutions provide complementary granularity: 1×1 gives a true global summary, 3×3 preserves coarse regional variation, and 7×7 retains finer spatial heterogeneity. The non-consecutive spacing also avoids using three highly redundant neighboring pooling scales; its empirical sensitivity is evaluated in Section 4.5. Here, $\mathbf{P}_r$ provides a spatial summary of the input features. For each pooled representation, per-channel spatial statistics are computed to capture first-order activation magnitude and second-order structural variation. Specifically, the spatial mean and variance within each channel are obtained as

$$\boldsymbol{\mu}_r = \frac{1}{r^2}\sum_{h=1}^{r}\sum_{w=1}^{r}\mathbf{P}_r(:,:,h,w), \quad \boldsymbol{\mu}_r \in \mathbb{R}^{B\times C}, \tag{6}$$

$$\boldsymbol{\nu}_r = \frac{1}{r^2}\sum_{h=1}^{r}\sum_{w=1}^{r}\left(\mathbf{P}_r(:,:,h,w) - \boldsymbol{\mu}_r\right)^2, \quad \boldsymbol{\nu}_r \in \mathbb{R}^{B\times C}, \tag{7}$$

where $\boldsymbol{\mu}_r$ reflects the average value of each channel, while $\boldsymbol{\nu}_r$ measures the spatial heterogeneity of each channel. These multi-resolution statistics are concatenated along the channel dimension and projected to form the global scale descriptor:

$$\mathbf{G} = \mathcal{W}_g(\mathrm{Concat}_C[\boldsymbol{\mu}_1, \boldsymbol{\nu}_1, \boldsymbol{\mu}_3, \boldsymbol{\nu}_3, \boldsymbol{\mu}_7, \boldsymbol{\nu}_7]) \in \mathbb{R}^{B\times K}, \tag{8}$$

where $\mathrm{Concat}_C[\cdot]$ aggregates the six tensors along the channel axis to yield a feature vector of size $B \times 6C$, and $\mathcal{W}_g$ is a linear projection layer mapping the aggregated statistics onto K dimensions. The construction of the multi-resolution scale descriptor is illustrated in Figure 2.

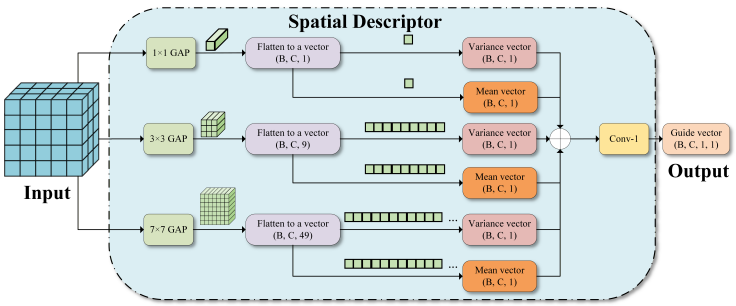


Figure 2. Spatial descriptor design for global scale guidance. The input feature map is first pooled at multiple spatial resolutions, then flattened into scale-specific statistical vectors and further represented by mean and variance descriptors. The resulting descriptors from different resolutions are finally integrated and projected to form a global guidance vector for scale-aware gated routing.

By combining both first- and second-order moments at multiple resolutions, the descriptor encodes complementary global-scale information without introducing heavy computational overhead. In addition, both the teacher and the student adopt the same descriptor topology and statistic configuration, ensuring that the image-level scale prior remains consistent across model capacities and enabling stable alignment of global routing behavior during distillation.

3.2.3. Spatial Bias Signal

The global scale descriptor and branch-response feedback provide scene-level scale priors and expert-response-dependent routing cues. However, their interaction may still be insufficient to preserve fine local spatial evidence, especially when global statistics are ambiguous or branch responses are weakly separated. Therefore, SAGRD introduces an additional spatial bias signal as a residual routing term. This term supplies local geometric and structural cues, anchoring the routing decision to the underlying spatial patterns of the feature map rather than relying solely on global scale tendencies or branch-level responses.

To model local geometric variation, SAGRD employs a deformable spatial routing branch based on Deformable Convolutional Networks v2 (DCNv2) [20]. By learning adaptive offsets, the routing module can adjust its spatial sampling pattern according to local structures. This property is particularly important for remote sensing scenes containing elongated ships, irregular vehicle clusters, or anisotropic object distributions. Given the input feature map $\mathbf{F} \in \mathbb{R}^{B\times C\times H\times W}$, the spatial routing signal is generated through a reduction–deformation–refinement–projection pipeline:

$$\mathbf{F}_{\text{red}} = \mathcal{W}_{\text{red}}(\mathbf{F}), \quad \mathbf{F}_{\text{dcn}} = \text{DCNv2}(\mathbf{F}_{\text{red}}), \quad \mathbf{Q} = \mathcal{W}_{\text{spat}}\big(\phi_{\text{loc}}(\mathbf{F}_{\text{dcn}})\big) \in \mathbb{R}^{B\times K\times H\times W}, \tag{9}$$

where $\mathcal{W}_{\text{red}}$ is a 1×1 convolution that reduces the channel dimension from C to an intermediate size C', ϕ_{loc} denotes a local convolutional refinement block followed by batch normalization and ReLU, and $\mathcal{W}_{\text{spat}}$ is the final 1×1 projection layer mapping the refined features onto K channels. The spatial signal $\mathbf{Q}$ thereby shares the same tensor shape $\mathbb{R}^{B\times K\times H\times W}$ as the branch-response tensor $\mathbf{R}$, enabling direct addition in the unified routing formulation.

In the teacher–student framework, the spatial routing topology is kept aligned by using the same reduction–DCNv2–refinement–projection pipeline in both models, so that their routing distributions remain comparable for gate distillation. The capacity of this branch is asymmetric: the teacher uses a wider C' and refines the DCNv2 features with a standard convolutional block, whereas the student uses a smaller C' and replaces the post-DCNv2 refinement with a depthwise-separable block. This design preserves deformable spatial routing in both models while reducing the routing overhead of the lightweight student. A controlled comparison between depthwise-separable and standard post-DCNv2 refinement is provided in Section 4.5.

3.2.4. Unified Routing Formulation

After obtaining the three routing signals from Equations (4), (8), and (9), SAGRD combines them into unified routing logits:

$$\mathbf{Z} = \mathbf{G} \odot \mathbf{R} + \lambda_{\text{sp}}\mathbf{Q}, \quad \mathbf{Z} \in \mathbb{R}^{B\times K\times H\times W}, \tag{10}$$

where the global scale descriptor $\mathbf{G} \in \mathbb{R}^{B\times K}$ is reshaped to $\mathbb{R}^{B\times K\times 1\times 1}$ and broadcast over the spatial dimensions to match $\mathbf{R}$, $\odot$ denotes element-wise multiplication, and λ_{sp} is the spatial routing coefficient that balances the contribution of local structural bias against global scale modulation. The multiplicative interaction between $\mathbf{G}$ and $\mathbf{R}$ enables condi-

tional scale modulation: the global scale inductive bias selectively amplifies or suppresses branch-response patterns according to image-level scale tendencies. Meanwhile, the spatial routing signal $\mathbf{Q}$ is incorporated through additive fusion to preserve local structural details and avoid excessive suppression of spatial variations.

The routing weights are then obtained by applying Softmax normalization along the branch dimension at each spatial location:

$$A_{b,k,h,w} = \frac{\exp(Z_{b,k,h,w}/T)}{\sum_{j=1}^{K} \exp(Z_{b,j,h,w}/T)}, \tag{11}$$

where T denotes the routing temperature. Let $\mathbf{A}_k \in \mathbb{R}^{B\times 1\times H\times W}$ denote the routing weight map of the k-th branch, with each element given by $A_{b,k,h,w}$. The final routed feature is computed by aggregating the expert responses weighted by their corresponding gate maps:

$$\mathbf{F}_{\text{route}} = \mathcal{W}_{\text{agg}}\left(\sum_{k=1}^{K} \mathbf{A}_k \odot \mathbf{B}_k\right), \tag{12}$$

where $\mathcal{W}_{\text{agg}}$ is a 1×1 fusion projection layer and the broadcast from $\mathbf{A}_k$'s single channel to $\mathbf{B}_k$'s C' channels is performed implicitly within the element-wise product. Through the joint modeling of global scale tendency, expert-response behavior, and local spatial structure, the proposed mechanism transforms conventional static multi-scale fusion into content-dependent receptive-field weighting conditioned on scene characteristics. The complete routing procedure is summarized in Algorithm 1.

Algorithm 1 Scale-aware gated routing mechanism.

Input: Input feature map $\mathbf{F} \in \mathbb{R}^{B\times C\times H\times W}$, number of branches K, routing temperature T, spatial coefficient λ_{sp}, gradient scale α.
Output: Routed feature $\mathbf{F}_{\text{route}}$.

Step 1: Construction of multi-scale expert branches.
for $k = 1$ **to** K **do**
  $\mathbf{B}_k \leftarrow \mathcal{E}_k(\mathbf{F})$
  $\tilde{\mathbf{B}}_k \leftarrow \text{GradScale}(\mathbf{B}_k; \alpha)$ {Gradient-decoupled feedback}
end for
Compute branch set $\tilde{\mathcal{B}} = \{\tilde{\mathbf{B}}_1, \tilde{\mathbf{B}}_2, \ldots, \tilde{\mathbf{B}}_K\}$.

Step 2: Generation of routing signals.
Obtain multi-resolution pooled representations $\{\mathbf{P}_r\}_{r\in\{1,3,7\}}$ by adaptive average pooling;
Compute per-channel spatial means $\{\boldsymbol{\mu}_r\}$ and variances $\{\boldsymbol{\nu}_r\}$ by Equations (6) and (7);
$\mathbf{G} \leftarrow \mathcal{W}_g(\text{Concat}_C[\boldsymbol{\mu}_1, \boldsymbol{\nu}_1, \boldsymbol{\mu}_3, \boldsymbol{\nu}_3, \boldsymbol{\mu}_7, \boldsymbol{\nu}_7])$;
$\mathbf{R} \leftarrow \mathcal{W}_r(\text{Concat}_C[\tilde{\mathbf{B}}_1, \tilde{\mathbf{B}}_2, \ldots, \tilde{\mathbf{B}}_K])$;
$\mathbf{F}_{\text{red}} \leftarrow \mathcal{W}_{\text{red}}(\mathbf{F})$;
$\mathbf{F}_{\text{dcn}} \leftarrow \text{DCNv2}(\mathbf{F}_{\text{red}})$;
$\mathbf{Q} \leftarrow \mathcal{W}_{\text{spat}}(\phi_{\text{loc}}(\mathbf{F}_{\text{dcn}}))$.

Step 3: Routing weight prediction.
Compute routing logits $\mathbf{Z} \leftarrow \mathbf{G} \odot \mathbf{R} + \lambda_{\text{sp}}\mathbf{Q}$;
Compute routing weight maps $\{\mathbf{A}_k\}_{k=1}^{K}$ with $A_{b,k,h,w} = \frac{\exp(Z_{b,k,h,w}/T)}{\sum_{j=1}^{K}\exp(Z_{b,j,h,w}/T)}$.

Step 4: Routed feature aggregation.
$\mathbf{F}_{\text{route}} \leftarrow \sum_{k=1}^{K} \mathbf{A}_k \odot \mathbf{B}_k$;
$\mathbf{F}_{\text{route}} \leftarrow \mathcal{W}_{\text{agg}}(\mathbf{F}_{\text{route}})$.
return $\mathbf{F}_{\text{route}}$

3.3. Context Refinement and Cross-Scale Fusion

SAGRD adopts a three-hierarchy encoding structure to capture multi-level representations at different spatial resolutions. The detail layer and the semantic layer both employ the scale-aware gated routing mechanism presented in Section 3.2, while the context layer handles the deepest low-resolution feature for holistic scene understanding. Concretely, let $\mathbf{X}_d$, $\mathbf{X}_s$, and $\mathbf{X}_c$ denote the shallow high-resolution feature, the mid-level semantic feature, and the deepest context feature extracted from the backbone, respectively. The detail and semantic routers generate scale-adaptive features as

$$\mathbf{F}_d = \mathrm{GatedRouter}(\mathbf{X}_d), \qquad \mathbf{F}_s = \mathrm{GatedRouter}(\mathbf{X}_s), \tag{13}$$

where $\mathrm{GatedRouter}(\cdot)$ denotes the unified routing formulation in Equations (10)–(12). The two routers share the same topology and differ only in their input channel counts and spatial resolutions: the detail router preserves fine-grained spatial information at high resolution, whereas the semantic router captures richer semantic patterns at a coarser resolution.

Because the context feature $\mathbf{X}_c$ resides at the lowest resolution and already has a substantially enlarged effective receptive field, SAGRD applies a dedicated context refinement module rather than another multi-branch router. The context stage therefore has no K-branch gate distribution to regularize. Adding parallel dilated experts at this stage would provide limited additional receptive-field diversity while increasing computation and memory cost. This module employs a MobileViT block $\mathcal{M}(\cdot)$ to model long-range dependencies and then fuses the global context back into the original feature via a residual convolutional block:

$$\mathbf{F}_c = \phi\Big(\mathcal{W}_{\mathrm{cx}}\big(\mathbf{X}_c + \mathcal{M}(\mathbf{X}_c)\big) + \mathbf{X}_c\Big), \tag{14}$$

where $\mathcal{W}_{\mathrm{cx}}$ is a 3×3 convolution followed by batch normalization, and ϕ denotes the ReLU6 function. In the teacher–student framework, the context refinement topology remains aligned, both models adopt MobileViT blocks for global modeling while the capacity is asymmetric: the teacher uses a larger transformer dimension, more attention heads, deeper transformer layers, and a larger MLP expansion ratio, whereas the student uses a compact MobileViT configuration to preserve global context modeling under a lightweight computational budget.

After routing and refinement, the three multi-level features occupy different spatial resolutions. To align them for joint processing, the semantic and context features are bilinearly upsampled to the spatial resolution of the detail feature:

$$\hat{\mathbf{F}}_s = \mathrm{Upsample}(\mathbf{F}_s; H_d, W_d), \qquad \hat{\mathbf{F}}_c = \mathrm{Upsample}(\mathbf{F}_c; H_d, W_d), \tag{15}$$

where (H_d, W_d) denotes the spatial size of $\mathbf{F}_d$. The aligned features are concatenated along the channel dimension and processed by a lightweight cross-scale semantic fusion module:

$$\begin{aligned} \mathbf{F}_{\mathrm{cat}} &= \mathrm{Concat}_C\big[\mathbf{F}_d, \hat{\mathbf{F}}_s, \hat{\mathbf{F}}_c\big], \\ \mathbf{F}_{\mathrm{fuse}} &= \phi\Big(\mathcal{W}_{\mathrm{proj}}\Big(\mathrm{SCConv}\big(\mathcal{W}_{\mathrm{mix}}(\mathcal{W}_{\mathrm{fus}}(\mathbf{F}_{\mathrm{cat}}))\big)\Big) + \mathcal{W}_{\mathrm{skip}}(\mathbf{F}_{\mathrm{cat}})\Big), \end{aligned} \tag{16}$$

where $\mathbf{F}_{\mathrm{cat}}$ denotes the channel-wise concatenation of the three aligned features; $\mathcal{W}_{\mathrm{fus}}$ and $\mathcal{W}_{\mathrm{mix}}$ are 1×1 and 3×3 convolutions, respectively, each followed by batch normalization and ReLU6; $\mathrm{SCConv}(\cdot)$ is the spatial–channel reconstruction block comprising a spatial reconstruction unit (SRU) and a channel reconstruction unit (CRU); $\mathcal{W}_{\mathrm{proj}}$ and $\mathcal{W}_{\mathrm{skip}}$ are 1×1 projection and shortcut convolutions, respectively, each followed by batch normalization (without activation); and ϕ denotes the ReLU6 activation applied to the residual sum. This design allows the network to jointly preserve fine local details from the detail branch,

mid-level semantics from the semantic branch, and global contextual guidance from the context branch.

Finally, the fused feature $\mathbf{F}_{\text{fuse}}$ is fed into a lightweight density estimation head composed of depthwise separable convolutions to produce the predicted density map $\mathbf{D}$. A Softplus activation is applied at the output layer to ensure non-negative density values while maintaining stable gradient propagation. The final object count is obtained by spatially integrating all values in $\mathbf{D}$.

3.4. Routing-Oriented Knowledge Distillation

To improve lightweight counting performance, SAGRD adopts a teacher–student distillation framework. Unlike conventional output distillation that only transfers final predictions, SAGRD additionally distills routing behavior from the teacher model. The teacher is kept fixed during this stage so that both density and routing supervision remain stationary; the present work does not claim that this offline protocol is inherently superior to joint teacher–student optimization.

Given an input image, the teacher and student models produce density maps $\mathbf{D}_t$ and $\mathbf{D}_s$, respectively. Following the standard output-distillation paradigm, density-map consistency is enforced via a normalized squared error:

$$L_{\text{dens}}^{\text{kd}} = \frac{1}{N} \left\| \mathbf{D}_s - \mathbf{D}_t \right\|_2^2, \tag{17}$$

where N denotes the number of spatial pixels in the density map. The normalization by N makes the loss invariant to input resolution and therefore stable across different training scales.

Beyond density supervision, SAGRD transfers routing knowledge at the image level. Recall from Section 3.3 that the network contains two routing stages: the detail pyramid ($l = d$) and the semantic pyramid ($l = s$). At each stage l, the teacher and student routers generate per-pixel branch distributions $\mathbf{A}_t^{(l)}, \mathbf{A}_s^{(l)} \in \mathbb{R}^{B \times K \times H_l \times W_l}$, where each spatial location stores a K-dimensional probability vector obtained by the Softmax in Equation (11). To avoid over-constraining the lightweight student with dense pixel-wise alignment, these spatial distributions are aggregated into image-level routing vectors by averaging over all spatial positions:

$$\bar{a}_{t,k}^{(l)} = \frac{1}{H_l W_l} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} A_{t,b,k,h,w}^{(l)}, \qquad \bar{a}_{s,k}^{(l)} = \frac{1}{H_l W_l} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} A_{s,b,k,h,w}^{(l)}, \tag{18}$$

where the subscripts b and k index the batch sample and the branch, respectively. By construction, $\sum_{k=1}^{K} \bar{a}_{t,k}^{(l)} = \sum_{k=1}^{K} \bar{a}_{s,k}^{(l)} = 1$, so each $\bar{\boldsymbol{a}}^{(l)}$ forms a valid probability distribution over the K receptive-field branches. The routing distillation loss is then defined as the sum of Kullback–Leibler divergences between the teacher and student distributions:

$$L_{\text{gate}}^{\text{kd}} = \lambda_{\text{branch}} \sum_{l \in \{d,s\}} \sum_{k=1}^{K} \bar{a}_{t,k}^{(l)} \log \frac{\bar{a}_{t,k}^{(l)}}{\bar{a}_{s,k}^{(l)}}, \tag{19}$$

where λ_{branch} is a branch-level magnification that compensates for the smaller magnitude of the routing distribution loss relative to the density loss. The summation explicitly shows that the KL is computed along the branch dimension k, with the teacher acting as the target distribution.

The final distillation objective combines density-map supervision and routing-distribution supervision through a single gating coefficient λ_g:

$$L_{\text{kd}} = \lambda_d L_{\text{dens}}^{\text{kd}} + \lambda_g L_{\text{gate}}^{\text{kd}}. \tag{20}$$

Here, λ_d and λ_g control the relative contributions of the two distillation terms, while the internal weighting λ_{branch} is absorbed into $L_{\text{gate}}^{\text{kd}}$ to balance the per-branch gradient magnitudes. By matching the teacher's global branch preferences, the student learns structured receptive-field adaptation strategies, which significantly strengthens teacher-to-student guidance under limited model capacity.

3.5. Routing Stabilization

Although the branch-response feedback mechanism in Equation (4) improves expert-aware scale selection, it also introduces strong optimization coupling between the router and the expert branches. Specifically, the gradient of the routing loss with respect to the projection layer $\mathcal{W}_r$ propagates back through the concatenation of $\{\mathbf{B}_k\}_{k=1}^{K}$, creating a positive feedback loop: the router assigns higher weights to better-performing branches, which in turn receive stronger gradients and further dominate, causing premature collapse toward a single expert before sufficient scale specialization is established. To mitigate this coupling, SAGRD introduces three complementary routing stabilization strategies: gradient-decoupled branch feedback, temperature annealing, and complexity-conditioned gate regularization.

3.5.1. Gradient-Decoupled Branch Feedback

The branch-response feedback mechanism enables the routing module to observe the current activation behavior of different receptive-field experts. However, directly propagating routing gradients through the concatenated branch activations may destabilize expert learning. Strong routing supervision can force expert branches to optimize excessively for routing preference rather than for discriminative feature representation.

To alleviate this problem, SAGRD applies gradient-decoupled scaling to the branch-feedback path. For each expert response $\mathbf{B}_k$, the stabilized feature used for routing prediction is defined as

$$\tilde{\mathbf{B}}_k = \text{GradScale}(\mathbf{B}_k; \alpha), \tag{21}$$

where $\text{GradScale}(\cdot; \alpha)$ denotes a straight-through gradient-scaling operator. In the forward pass, it acts as an identity mapping, i.e., $\tilde{\mathbf{B}}_k = \mathbf{B}_k$, whereas in the backward pass, it scales the gradient by a constant factor $\alpha \in [0, 1]$:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{B}_k} = \alpha \cdot \frac{\partial \mathcal{L}}{\partial \tilde{\mathbf{B}}_k}. \tag{22}$$

In practice, we set $\alpha = 0.05$, so that the router can still observe the original expert-response information for adaptive scale selection, while preventing unstable routing gradients from dominating expert optimization. By weakening the router-to-expert gradient interference, this strategy improves expert diversity and stabilizes the co-optimization of the router and the branches.

3.5.2. Temperature Annealing Strategy

Another critical issue in dynamic routing is the early-stage winner-takes-all effect. In Equation (11), the routing weights are obtained by applying a per-pixel Softmax over the K-dimensional routing logits. At the beginning of training, the routing logits are numerically uncalibrated and can exhibit spurious margins among branches. Because the

Softmax operator exponentially amplifies relative differences at low temperatures, a small logit gap at $T \approx 1$ forces the gate to collapse onto a single branch, depriving the remaining experts of gradient feedback. To prevent such premature specialization, SAGRD employs temperature annealing to control the sharpness, and hence the entropy, of the Softmax routing distribution without introducing an additional entropy-loss term.

Concretely, the gating temperature T is initialized to a high value $T_{\max}$ and gradually annealed toward $T_{\min}$ according to a cosine schedule during the first N_{anneal} epochs. Denoting the current epoch by t and the annealing start epoch by t_{start}, the temperature is defined as

$$T(t) = \begin{cases} T_{\max}, & t < t_{\text{start}}, \\ T_{\min} + \dfrac{T_{\max} - T_{\min}}{2}\left(1 + \cos\left(\pi \dfrac{t - t_{\text{start}}}{N_{\text{anneal}}}\right)\right), & t_{\text{start}} \le t < t_{\text{start}} + N_{\text{anneal}}, \\ T_{\min}, & t \ge t_{\text{start}} + N_{\text{anneal}}. \end{cases} \tag{23}$$

In our implementation, we set $T_{\max} = 30.0$, $T_{\min} = 1.0$, $N_{\text{anneal}} = 10$, and $t_{\text{start}} = 0$. At $T_{\max} = 30.0$, the Softmax distribution approaches uniform, so that all expert branches receive comparable gradient magnitudes and are encouraged to learn diverse receptive-field representations during the exploration phase. As the cosine curve smoothly transitions toward $T_{\min}$, the routing distribution gradually sharpens, allowing the gate to confidently select the most appropriate receptive-field branch for each spatial location once the routing logits are well-calibrated. Compared with a piece-wise linear decay, the cosine schedule provides a gentler rate of change at the beginning and end of annealing while accelerating the transition in the middle, which empirically yields more stable co-training of the router and the experts.

The annealed temperature is shared across both the detail and semantic routing pyramids by dynamically updating the gate temperature attribute of each ScaleAwareAdaptivePyramid module at the beginning of every training epoch. After annealing completes, the temperature remains at $T_{\min}$ for the remaining epochs, ensuring that the final routing behavior is governed by a standard low-temperature Softmax gating in Equation (11).

3.6. Training Loss

The overall optimization objective of SAGRD consists of four components: density regression loss, count loss, routing-oriented distillation loss, and gate regularization loss. The complete objective is designed to jointly optimize density estimation accuracy, adaptive routing behavior, and branch diversity.

Given the predicted density map $\mathbf{D}_s$ and the ground-truth density map $\mathbf{D}_{\text{gt}}$, the supervised density regression loss is defined as

$$L_{\text{dens}} = \frac{1}{N}\left\|\mathbf{D}_s - \mathbf{D}_{\text{gt}}\right\|_2^2, \tag{24}$$

where N denotes the number of pixels in the density map. To further preserve global counting consistency, an additional count loss is introduced:

$$L_{\text{count}} = \left|C_s - C_{\text{gt}}\right|, \tag{25}$$

where $C_s = \sum_{h=1}^{H}\sum_{w=1}^{W}\mathbf{D}_s(:,:,h,w)$ and $C_{\text{gt}} = \sum_{h=1}^{H}\sum_{w=1}^{W}\mathbf{D}_{\text{gt}}(:,:,h,w)$ are the predicted and ground-truth object counts obtained by spatial integration over the density maps, respectively.

In addition, the routing-oriented distillation loss defined in Equation (20) is incorporated into the final objective:

$$L_{\mathrm{kd}} = \lambda_d L_{\mathrm{dens}}^{\mathrm{kd}} + \lambda_g L_{\mathrm{gate}}^{\mathrm{kd}}, \tag{26}$$

where $L_{\mathrm{dens}}^{\mathrm{kd}}$ and $L_{\mathrm{gate}}^{\mathrm{kd}}$ denote the density-map distillation and routing-distribution distillation losses, respectively.

Although the proposed gated routing mechanism enables input-dependent receptive-field weighting, the router may still degenerate into single-pattern behavior, where images with different structural complexity activate similar dominant branches. Rather than adding an explicit entropy-maximization loss that would encourage uniformly distributed routing weights, we introduce a complexity-conditioned batch-wise gate diversity regularization. The goal is not to make every sample use all branches equally, but to discourage structurally different images from sharing the same dominant routing preference.

Let $\mathbf{A}^{(l)} \in \mathbb{R}^{B\times K\times H_l\times W_l}$ denote the routing weight tensor at router level $l \in \{d, s\}$, where d and s correspond to the detail and semantic routing pyramids, respectively. Since the regularization is applied to image-level routing preference, the spatial routing weights are first averaged over all spatial locations:

$$\bar{a}_{i,k}^{(l)} = \frac{1}{H_l W_l} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} A_{i,k,h,w}^{(l)}, \tag{27}$$

where $\bar{\mathbf{a}}_i^{(l)} = [\bar{a}_{i,1}^{(l)}, \ldots, \bar{a}_{i,K}^{(l)}]^\top \in \mathbb{R}^K$ satisfies $\sum_{k=1}^K \bar{a}_{i,k}^{(l)} = 1$, and $i \in \{1, \ldots, B\}$ indexes the sample in a batch.

To estimate the structural complexity of each input image, we compute an edge-density-based complexity score. Given the input image I_i, it is first converted to a grayscale image:

$$\bar{\mathbf{G}}_i = 0.299\, I_i^{\mathrm{R}} + 0.587\, I_i^{\mathrm{G}} + 0.114\, I_i^{\mathrm{B}}. \tag{28}$$

A valid-region mask is generated to suppress invalid padded regions:

$$\mathbf{M}_i(x) = \mathbb{1}[\bar{\mathbf{G}}_i(x) > \theta_b], \tag{29}$$

where x indexes the spatial pixel and θ_b is the brightness threshold. Sobel filters are then applied to compute the edge magnitude:

$$\mathbf{E}_i(x) = \sqrt{\left[(\bar{\mathbf{G}}_i * S_x)(x)\right]^2 + \left[(\bar{\mathbf{G}}_i * S_y)(x)\right]^2 + \epsilon}\,, \tag{30}$$

where S_x and S_y denote the horizontal and vertical Sobel kernels, $*$ denotes convolution, and ϵ is a small constant for numerical stability. The image complexity score is defined as the edge density within the valid region:

$$c_i = \frac{\sum_x \mathbb{1}[\mathbf{E}_i(x) > \theta_e] \cdot \mathbf{M}_i(x)}{\sum_x \mathbf{M}_i(x) + \epsilon}, \tag{31}$$

where θ_e is the edge threshold. A larger c_i indicates richer local structures and higher scene complexity.

To focus on the dominant routing preference, the averaged gate vector is sharpened using a low-temperature Softmax:

$$\mathbf{p}_i^{(l)} = \mathrm{Softmax}\left(\frac{\bar{\mathbf{a}}_i^{(l)}}{\tau_b}\right), \tag{32}$$

where τ_b is the batch-diversity temperature and the Softmax is applied along the branch dimension. The similarity between two samples is measured by the inner product of their sharpened gate distributions:

$$s_{ij}^{(l)} = \left(\mathbf{p}_i^{(l)}\right)^\top \mathbf{p}_j^{(l)}. \tag{33}$$

A large $s_{ij}^{(l)}$ indicates that the two samples activate similar dominant receptive-field branches. The complexity-conditioned gate diversity loss for router level l is then formulated as

$$L_{\text{div}}^{(l)} = \frac{2}{B(B-1)} \sum_{1 \le i < j \le B} |c_i - c_j| \cdot s_{ij}^{(l)}, \tag{34}$$

where B is the batch size. This loss penalizes sample pairs that have large structural-complexity differences but similar dominant routing preferences. For samples with similar complexity, the weighting term $|c_i - c_j|$ is small, and therefore selecting similar branches is not strongly penalized. The final gate regularization term is applied to both detail and semantic routers:

$$L_{\text{gate}} = L_{\text{div}}^{(d)} + L_{\text{div}}^{(s)}. \tag{35}$$

The regularizer is defined only for the detail and semantic routers because these are the only stages that produce K-branch routing distributions; the context stage uses direct context refinement and therefore has no gate vector to regularize. Finally, the overall training objective of SAGRD is formulated as

$$L_{\text{total}} = L_{\text{dens}} + \lambda_c L_{\text{count}} + \lambda_{\text{kd}} L_{\text{kd}} + \lambda_r L_{\text{gate}}, \tag{36}$$

where λ_c, λ_{kd}, and λ_r denote the balancing coefficients for count supervision, distillation supervision, and gate regularization, respectively.

4. Experiments and Results

4.1. Implementation Details

Our experimental training environment uses Ubuntu 22.04, Python version 3.12, PyTorch version 2.8, CUDA 12.8, and an NVIDIA RTX PRO 6000. Inference-speed experiments are conducted on an NVIDIA Jetson Orin Nano (8 GB). The platform provides 67 INT8 TOPS, 1024 CUDA cores, a 6-core Arm Cortex-A78AE CPU, 8 GB RAM, and 512 GB NVMe storage.

Training is performed in two stages. The teacher is first optimized independently, and its checkpoint is selected according to the lowest MAE on a validation subset separated from the training data; the test split is not used for checkpoint selection or hyperparameter tuning. The selected teacher is then frozen throughout student training and provides stationary density and routing targets, following an offline distillation protocol. The student is initialized with ImageNet-pretrained MobileNetV3 weights. Joint teacher–student fine-tuning is not used in the present study because it would turn the teacher routing distribution into a moving target and introduce a different co-optimization protocol.

During model training, the model is trained for 200 epochs using SGD with an initial learning rate of 2×10^{-4}, momentum of 0.9, weight decay of 1×10^{-3}, and a batch size of 8, while the number of data-loading workers is set to 8. The ground-truth density loss is applied with unit weight. The count loss weight is set to $\lambda_c = 0.5$, with a 5-epoch warm-up. For knowledge distillation, the global distillation scale is set to $\lambda_{\text{kd}} = 0.5$, with a 5-epoch ramp-up. The output density distillation weight and gate distillation weight are set to $\lambda_d = 0.1$ and $\lambda_g = 0.5$, respectively, and the detail and semantic gate distillation terms are both scaled by $\lambda_{\text{branch}} = 4$. The batch-wise gate diversity regularization weight is set to $\lambda_r = 5$ and ramped up over the first 5 epochs. These coefficients are empirically selected

within the tested ranges; their sensitivity and the separate contributions of density and routing distillation are evaluated in Section 4.5. The routing temperature is cosine-annealed from $T_{\max} = 30.0$ to $T_{\min} = 1.0$ during the first $N_{\text{anneal}} = 10$ epochs and is fixed at $T_{\min}$ thereafter, consistent with the schedule defined in the Routing Stabilization section.

For density-map generation, all annotations are represented as point-based supervision. Benchmark-provided point annotations are retained directly, whereas bounding-box or oriented-quadrilateral annotations are converted to their geometric centers. These reference points are not re-estimated visual centroids; for elongated, irregular, or partially visible objects, the geometric annotation center may therefore differ slightly from the visually perceived center. Resizing and integer-coordinate discretization can introduce additional small offsets. Given the resulting points $\{p_i\}_{i=1}^{N}$, the ground-truth density map is generated by Gaussian convolution. The kernel size is set to 10 for RSOC Building and to 5 for the other datasets. Gaussian spreading preserves the unit mass of each annotation and makes modest local offsets less influential on the image-level count, although such ambiguity can still reduce peak-localization fidelity, especially for tiny objects.

For RSOC images exceeding 768×1024, the spatial resolution is capped at 768×1024 before cropping. This preprocessing bounds the size of intermediate feature maps and therefore limits activation-memory usage and computational cost during batch training; convolutional parameter matrices themselves are independent of the input resolution. This efficiency-oriented preprocessing is not an intrinsic requirement of SAGRD and introduces a trade-off: downsampling removes high-frequency details, reduces the apparent pixel size of small objects, and may merge nearby responses in dense regions. It can therefore weaken the local evidence available to the detail router and reduce scale fidelity for tiny targets. Random cropping and horizontal flipping are subsequently applied for augmentation. The crop size is 256×256 for RSOC Building and ShanghaiTech Part A and 512×512 for the other datasets. The dataset statistics and train/test splits are summarized in Table 2.

Table 2. Dataset statistics used for evaluation. Each dataset or object category is trained and tested independently with strictly separated training and test splits.

Dataset	Category	Images	Train/Test	Average Resolution	Annotation	Count Statistics			
						Total	Min	Avg.	Max
RSOC [2]	Building	2468	1205/1263	512×512	Center point	76,215	15	30.88	142
RSOC [2]	Small vehicle	280	222/58	2473×2339	Oriented box	148,838	17	531.56	8531
RSOC [2]	Large vehicle	172	108/64	1552×1573	Oriented box	16,594	12	96.48	1336
RSOC [2]	Ship	137	97/40	2558×668	Oriented box	44,892	50	327.68	1661
CARPK [1]	Car	1448	989/459	1280×720	Bounding box	89,777	1	62.00	188
PUCPR+ [16]	Car	125	100/25	1280×720	Bounding box	16,915	0	135.00	331
ShanghaiTech Part A [14]	Crowd	482	300/182	589×868	Point annotation	241,667	–	501.00	3139
ShanghaiTech Part B [14]	Crowd	716	400/316	768×1024	Point annotation	88,488	–	123.00	578

4.2. Evaluation Metrics

Following common practice in object counting, mean absolute error (MAE) and root mean squared error (RMSE) are used as the main evaluation metrics. Given N test images, the ground-truth count of the i-th image is denoted as C_i, and the predicted count is denoted as $\hat{C}_i$. MAE and RMSE are defined as

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\hat{C}_i - C_i|, \tag{37}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{C}_i - C_i)^2}. \tag{38}$$

MAE measures the average absolute counting error, while RMSE penalizes large errors more strongly and reflects the robustness of a model on difficult samples.

4.3. Comparison with State-of-the-Art Methods

To evaluate the competitiveness of SAGRD, we compare it with representative object counting and remote sensing counting methods, the comparison is summarized in Table 3.

Table 3. Comparison with representative state-of-the-art object counting methods on remote sensing object counting datasets. The compared methods are mainly reported on the RSOC benchmark, while SAGRD results follow the dataset-wise supervised evaluation protocol used in this work. Bold values indicate the best reported value in the corresponding column.

Method	Params	FLOPs	Building		Large Vehicle		Ship		Small Vehicle	
			MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
CSRNet [15]	16.26 M	325.30 G	8.00	11.78	34.10	46.42	240.01	394.81	443.72	1252.22
CAN [3]	18.18 M	–	9.12	13.38	34.56	49.63	282.69	423.44	457.36	1260.39
SFANet [25]	–	–	8.18	11.75	29.04	47.01	201.61	332.87	435.29	1284.15
ASPDNet [2]	22.70 M	465.75 G	7.59	10.66	18.76	31.06	193.83	318.95	433.23	1238.61
PSGCNet [19]	27.51 M	385.79 G	7.54	10.52	**11.00**	**17.65**	74.91	112.11	157.55	245.31
eFreeNet [24]	–	–	5.62	7.69	14.55	19.77	65.34	85.45	195.86	463.62
MobileCount [30]	3.40 M	6.15 G	7.72	11.90	18.50	30.4	73.2	100.2	316.02	598.58
LMSFFNet [21]	4.58 M	14.90 G	6.33	9.38	12.74	27.13	49.47	85.03	141.7	273.0
LEDCrowdNet [32]	2.02 M	–	6.81	10.22	22.01	32.84	82.26	131.52	237.18	621.80
EdgeCount [22]	**0.12 M**	1.95 G	6.27	9.46	11.34	18.37	**36.38**	52.44	136.61	260.6
SAGRD-T	14.19 M	14.83 G	**4.678**	**6.33**	14.80	21.90	49.78	61.15	**114.03**	**210.17**
SAGRD-S	1.39 M	1.29 G	4.712	6.37	15.73	23.64	55.11	81.93	144.21	243.29

Specifically, SAGRD-T achieves the best overall performance on RSOC building with 4.678 MAE and 6.329 RMSE. Compared with LMSFFNet, SAGRD further reduces MAE from 6.33 to 4.678, corresponding to a 26.1% improvement. More importantly, the lightweight SAGRD-Student still achieves 4.712 MAE and 6.367 RMSE, remaining close to the teacher while using only 1.39 M parameters and 1.29 G FLOPs. Compared with EdgeCount, the student model achieves approximately 24.8% lower MAE and 32.7% lower RMSE, despite operating under a compact lightweight configuration. These results indicate that SAGRD can preserve strong counting capability even after aggressive model compression.

For the more challenging Large Vehicle, Ship, and Small Vehicle categories, the performance gap between methods becomes smaller and the overall counting difficulty increases substantially. On Large Vehicle, SAGRD-Student achieves 15.73 MAE and 23.64 RMSE, remaining competitive with existing strong methods. On Ship and Small Vehicle, although SAGRD still maintains reasonable performance, the advantage over previous methods becomes less stable, especially under dense small-object distributions and elongated-object scenes. In particular, the student model exhibits a noticeable degradation on Small Vehicle compared with the teacher, indicating that highly dense and visually ambiguous targets remain difficult for lightweight adaptive routing models. Nevertheless, the overall results show that SAGRD consistently maintains competitive counting accuracy across different scene types while operating under a significantly reduced computational budget. Representative qualitative results on the four RSOC categories are shown in Figure 3.

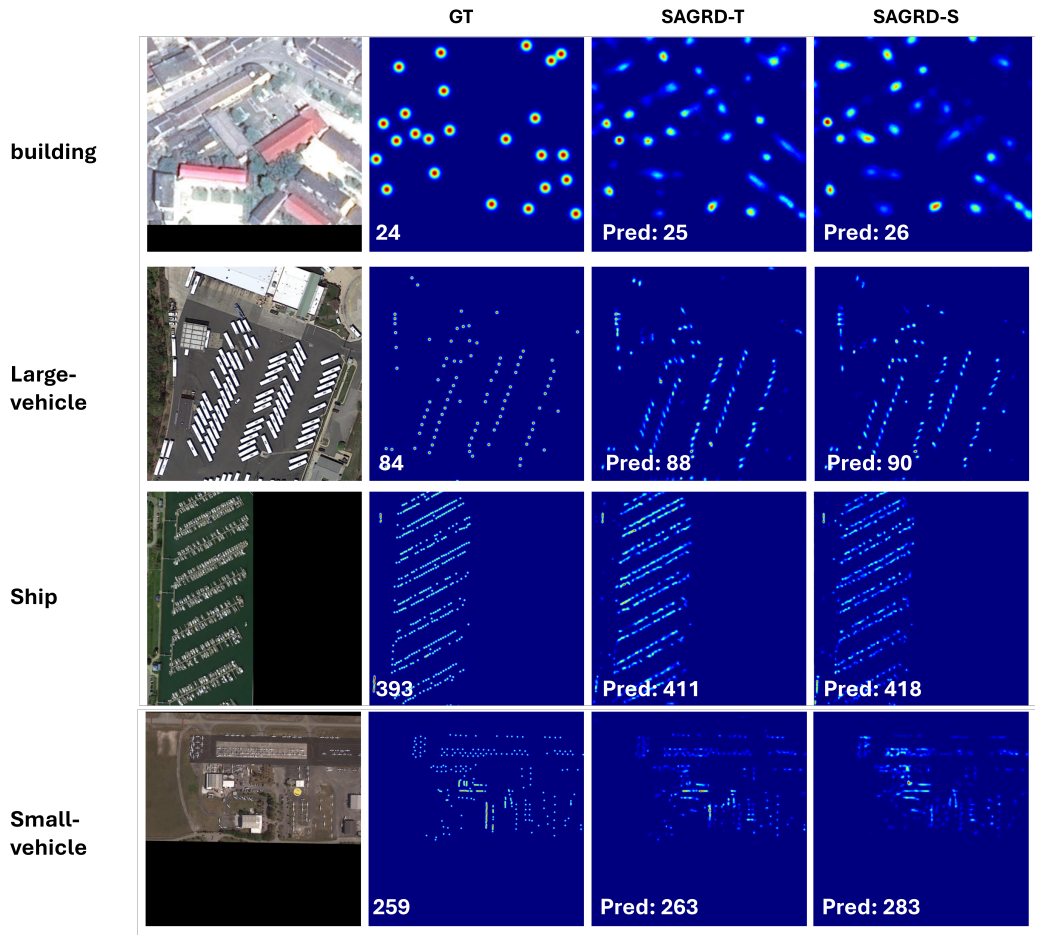


Figure 3. Visualizationresults on the RSOC dataset. From left to right: original images, ground-truth density maps (GT), predictions of SAGRD-T, and predictions of SAGRD-S; the four rows correspond to Building, Large Vehicle, Ship, and Small Vehicle. In the density maps, brighter cyan/yellow responses indicate higher density values, while dark blue indicates low or near-zero density. RSOC point annotations are retained when provided, while oriented-box annotations are converted to geometric centers before Gaussian density-map generation.

4.4. Other Dataset Results

CARPK and PUCPR+ are vehicle-counting benchmarks collected from parking-lot scenes, where cars appear with different densities, occlusions, and spatial layouts. Table 4 reports the comparison between SAGRD and representative counting methods on these two datasets. SAGRD-Teacher achieves the best overall performance, obtaining 5.67 MAE/7.37 RMSE on CARPK and 2.84 MAE/4.11 RMSE on PUCPR+. The lightweight SAGRD-Student also maintains competitive results, with 6.85 MAE/9.05 RMSE on CARPK and 3.10 MAE/4.41 RMSE on PUCPR+, remaining close to the teacher under a much smaller computational budget. These results show that the proposed routing-oriented distillation can preserve effective vehicle-counting performance in lightweight settings. Representative vehicle-counting results on CARPK are shown in Figure 4.

We also evaluate SAGRD on ShanghaiTech Parts A and B. ShanghaiTech Part A and Part B are widely used crowd-counting benchmarks, where Part A contains highly congested crowd scenes and Part B mainly consists of relatively sparse street-view scenes. Table 5 reports the experimental results on these two datasets. SAGRD-Teacher obtains 64.93 MAE/96.66 RMSE on Part A and 8.95 MAE/16.06 RMSE on Part B, while SAGRD-Student achieves 84.60 MAE/127.18 RMSE and 11.45 MAE/18.61 RMSE, respectively. The results are competitive with several classical counting methods but remain behind

recent crowd-specific approaches such as P2PNet and UOT. Representative crowd-counting results on ShanghaiTech Parts A and B are shown in Figure 5.

Table 4. Comparison on CARPK and PUCPR+ datasets. Bold values indicate the best result in the corresponding column.

Methods	CARPK		PUCPR+	
	MAE	**RMSE**	**MAE**	**RMSE**
LPN [1]	23.80	36.79	22.76	34.46
RetinaNet [39]	16.62	22.30	24.58	33.12
LEP [40]	51.83	–	15.17	–
MCNN [14]	39.10	43.30	21.86	29.53
CSRNet [15]	11.48	13.32	8.65	10.24
BL [4]	9.58	11.38	6.54	8.13
ASPDNet [2]	7.81	10.16	–	–
PSGCNet [19]	8.15	10.46	5.24	7.36
LMSFFNet [21]	7.05	9.03	4.49	6.21
EdgeCount [22]	7.12	9.44	3.39	5.15
SAGRD-T	**5.67**	**7.37**	**2.84**	**4.11**
SAGRD-S	6.85	9.05	3.10	4.41

Table 5. Comparison with state-of-the-art methods on ShanghaiTech (SHT-A, SHT-B). "Par. (M)" denotes the number of parameters in millions.

Methods	Par. (M)	SHT-A		SHT-B	
		MAE	**RMSE**	**MAE**	**RMSE**
CP-CNN [41]	68.40	73.6	106.4	20.1	30.1
Switch-CNN [42]	15.30	90.4	135.0	21.6	33.4
CSRNet [15]	16.26	68.2	115.0	10.6	16.0
BL [4]	21.50	61.5	103.2	7.5	12.6
UOT [5]	21.50	58.1	95.9	6.5	10.2
SUA-Fully [43]	15.85	66.9	125.6	12.3	17.9
UEPNet [44]	26.21	54.6	91.2	6.4	10.9
P2PNet [45]	18.34	52.7	85.1	6.3	9.9
DKPNet [46]	30.63	55.6	91.0	6.6	10.9
MCNN [14]	0.13	110.2	173.2	26.4	41.3
CMTL [47]	2.47	101.3	152.4	20.0	31.1
ACSCP [48]	5.10	75.7	102.7	17.2	27.4
TDF-CNN [49]	0.13	97.5	145.1	20.7	32.8
SANet [50]	0.91	75.3	122.2	10.5	17.9
PCC-Net [51]	0.55	73.5	102.7	11.0	19.0
LCNet [52]	0.86	93.3	149.0	15.3	25.2
C-CNN [53]	0.073	88.1	141.7	14.9	22.1
LMSFFNet [21]	4.58	85.85	139.9	9.2	15.1
SAGRD-T	14.19	64.93	96.66	8.95	16.06
SAGRD-S	1.39	84.60	127.18	11.45	18.61

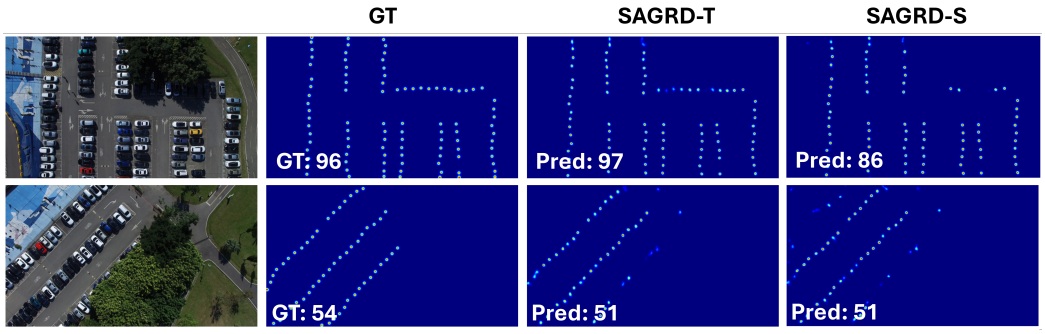


Figure 4. Visualization results of SAGRD-T and SAGRD-S on the CARPK dataset. From left to right: original images, ground-truth density maps (GT), predictions of SAGRD-T, and predictions of SAGRD-S. The predicted density maps show that both models can effectively localize and estimate vehicle distributions in parking scenes.

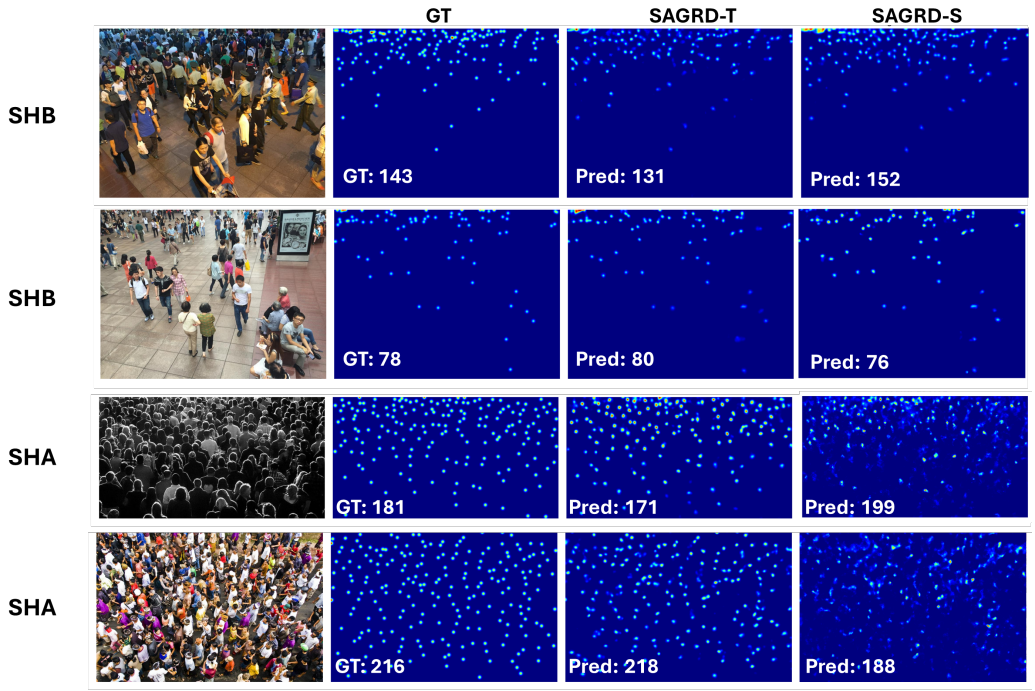


Figure 5. Visualization results on the ShanghaiTech Part A (SHA) and Part B (SHB) datasets. From left to right: original crowd images, ground-truth density maps (GT), predictions of SAGRD-T, and predictions of SAGRD-S. The visualization results demonstrate that the proposed models can effectively capture crowd distributions under both sparse and highly congested scenes while maintaining accurate counting performance.

These results indicate that SAGRD can be applied to general density-map-based counting, but its advantage is more evident in remote sensing object-counting scenarios. Highly crowded scenes in ShanghaiTech Part A involve severe occlusion, perspective distortion, and overlapping targets, which are different from the structured overhead patterns targeted by SAGRD. Therefore, the results on ShanghaiTech datasets are mainly reported as supplementary evidence of general counting capability, while the main focus of this work remains lightweight remote sensing object counting.

4.5. Ablation and Sensitivity Analysis

The ablation studies examine the performance changes when key components are removed or simplified, thereby verifying the necessity of each design in the proposed student model.

Table 6 shows that relying on a single routing signal is insufficient. The descriptor-only setting performs the worst, indicating that only global scale statistics cannot handle local spatial variation alone. The spatial-only and branch-only settings also remain limited because they lack either expert-response awareness or stable global scale guidance. The two-signal variants reduce the error but still miss one type of routing evidence. Therefore, the three routing signals are complementary, and removing any one of them weakens scale-aware routing.

Table 6. Ablation study of routing signals in the proposed student model. ✓ denotes that the corresponding routing signal is enabled, while – denotes disabled or absent.

Routing Signals			Params	MAE	RMSE
Desc.	Spat.	Branch			
✓	–	–	1.359 M	5.690	7.897
–	✓	–	1.373 M	5.467	7.349
–	–	✓	1.341 M	5.459	7.713
–	✓	✓	1.375 M	5.264	7.185
✓	–	✓	1.361 M	5.193	6.789
✓	✓	–	1.393 M	5.240	7.180

Table 7 shows the limitations of simplified fusion designs. The baseline model without multi-scale fusion has the highest error, confirming that a single-scale lightweight model cannot handle object-scale variation. Direct concatenation also performs worse than uniform fusion, suggesting that simply stacking multi-scale features may introduce redundant or inconsistent responses. Although uniform fusion is stronger than the baseline, it still lacks input-dependent branch selection. These results indicate that multi-scale features benefit from content-dependent soft routing rather than static fusion.

Table 7. Ablation study of base design and fusion strategies in the proposed student model. ✓ denotes that the corresponding setting is enabled, while – denotes disabled or absent.

Fusion Configuration			Params	MAE	RMSE
Baseline	Uniform	Concat			
✓	–	–	1.399 M	6.454	8.964
–	✓	–	1.395 M	5.102	6.764
–	–	✓	1.365 M	5.217	7.222

In this ablation, the corresponding loss weight is set to zero while the network architecture remains identical. Therefore, the parameter count remains unchanged across the distillation-loss ablation variants. Table 8 shows that removing distillation supervision degrades the student model. Without distillation, the compressed model cannot sufficiently approximate the teacher. Output KD alone transfers density predictions but ignores the teacher's routing behavior, while gate KD alone transfers scale-selection preferences without direct density-distribution guidance. The best result is obtained only when both losses are used, indicating that removing either term makes the transferred knowledge incomplete.

Table 8. Ablation study of distillation loss components in the proposed student model. ✓ denotes that the corresponding distillation loss is enabled, while – denotes disabled or absent.

Distillation		Params	MAE	RMSE
Gate KD	Output KD			
–	–	1.395 M	5.015	6.917
✓	✓	1.395 M	4.712	6.367
–	✓	1.395 M	5.162	6.937
✓	–	1.395 M	5.201	7.017

Table 9 shows the contribution of each proposed training stabilization strategy. When gradient-decoupled branch feedback is disabled, the MAE increases from 4.712 to 5.267, indicating that weakening direct gradient coupling between routing decisions and expert branches is essential to prevent unstable expert learning. Without temperature annealing, the MAE rises to 5.316 and the RMSE to 7.167, which confirms that high-temperature initialization followed by gradual sharpening is necessary to avoid premature branch specialization during early training. The most significant performance drop occurs when gate regularization is removed, with MAE increasing to 6.341 and RMSE to 8.809. This verifies that the gate diversity loss plays a critical role in preventing routing collapse and maintaining effective scale-adaptive feature selection. These results collectively demonstrate that all three stabilization components—gradient-decoupled feedback, temperature annealing, and gate regularization—are indispensable for reliable dynamic receptive-field routing in the lightweight student model.

Table 9. Ablation study of training stabilization strategies in the proposed student model. ✓ denotes that the corresponding strategy is enabled, while–denotes disabled or absent.

Training Stabilization			Params	MAE	RMSE
Grad. Scale	Temp. Anneal.	Gate Reg.			
✓	✓	✓	1.395 M	4.712	6.367
–	✓	✓	1.395 M	5.267	6.984
✓	–	✓	1.395 M	5.316	7.167
✓	✓	–	1.395 M	6.341	8.809

Sensitivity to Architectural and Loss Settings

We further evaluate several design choices questioned in the review. Unless otherwise stated, only the indicated component is changed and the remaining settings are kept identical.

Table 10 shows that every tested single-stage replacement degrades accuracy relative to $[2, 4, 6, 8]$. The results support retaining $[2, 4, 6, 8]$ as the best-performing configuration among the tested alternatives. We also observed lower training throughput when $d = 16$ was introduced in the teacher detail router, but this hardware-dependent observation is not used as the primary selection criterion.

The descriptor ablation in Table 11 shows that replacing $[1, 3, 7]$ with either $[1, 3, 5]$ or $[3, 5, 7]$ at the detail or semantic stage reduces accuracy. Retaining both the global 1×1 summary and the finer 7×7 summary therefore provides the most effective scale coverage among the tested settings.

Table 10. Sensitivity to dilation-rate configurations on RSOC Building. "None" denotes the original configuration, where both routers use $[2, 4, 6, 8]$. Bold values indicate the best result within each model group.

Model	Modified Router	Dilation Rates	MAE	RMSE
SAGRD-T	None	$[2, 4, 6, 8]$	**4.678**	**6.329**
SAGRD-T	Detail	$[1, 2, 4, 8]$	5.488	7.384
SAGRD-T	Semantic	$[1, 2, 4, 8]$	5.193	7.324
SAGRD-T	Detail	$[2, 4, 8, 16]$	5.400	7.318
SAGRD-T	Semantic	$[2, 4, 8, 16]$	5.535	7.623
SAGRD-S	None	$[2, 4, 6, 8]$	**4.712**	**6.367**
SAGRD-S	Detail	$[1, 2, 4, 8]$	5.616	7.686
SAGRD-S	Semantic	$[1, 2, 4, 8]$	5.591	7.360
SAGRD-S	Detail	$[2, 4, 8, 16]$	5.447	7.378
SAGRD-S	Semantic	$[2, 4, 8, 16]$	5.402	7.208

Table 11. Sensitivity of the student model to descriptor pooling scales on RSOC Building. Only the indicated router is changed. Bold values indicate the best result among the tested settings.

Modified Router	Pooling Sizes	MAE	RMSE
None	$[1, 3, 7]$	**4.712**	**6.367**
Detail	$[1, 3, 5]$	5.151	6.945
Semantic	$[1, 3, 5]$	5.249	7.037
Detail	$[3, 5, 7]$	5.438	7.522
Semantic	$[3, 5, 7]$	5.523	7.713

As shown in Table 12, replacing the depthwise-separable refinement with a standard convolution increases the parameter count from 1.395M to 1.807M (approximately 29.5%) without improving counting accuracy. This controlled comparison supports the lightweight refinement used by SAGRD-S. DCNv2 itself is retained in both variants so that deformable spatial sampling is unchanged.

Table 12. Effect of the post-DCNv2 refinement in SAGRD-S at 256×256 input resolution. Bold values indicate the better result between the two refinement variants.

Post-DCNv2 Refinement	Params	FLOPs	MAE	RMSE
Depthwise separable conv	**1.395 M**	**1.286 G**	**4.712**	**6.367**
Standard conv	1.807 M	2.159 G	4.879	6.499

The selected loss weights were determined empirically rather than assumed to be universally optimal. For gate diversity, $\lambda_r = 2$ produces overly uniform routing tendencies, whereas $\lambda_r = 10$ over-concentrates the routing distribution; $\lambda_r = 5$ gives the best accuracy among the tested values. The global distillation coefficient is comparatively robust over a moderate range, with $\lambda_{\text{kd}} = 0.5$ giving the best result in Table 13. For count supervision, reducing λ_c to 0.1 increases the teacher MAE to 5.521, whereas increasing it to 1.0 gives an MAE of 5.026 but produces visibly poorer density-map agreement with the ground truth; we therefore retain $\lambda_c = 0.5$ to balance count consistency and spatial density fidelity. The component ablation in Table 8 further shows that density-output and gate-routing distillation are complementary. We therefore use $\lambda_d = 0.1$ and $\lambda_g = 0.5$, while $\lambda_{\text{branch}} = 4$ is kept as a fixed magnitude compensation for the smaller routing KL term.

Table 13. Sensitivity to the main loss-balancing coefficients on RSOC Building. Bold values indicate the best result within each coefficient group.

Coefficient	Value	MAE	RMSE
λ_r	2	5.032	6.926
λ_r	**5**	**4.712**	**6.367**
λ_r	10	5.157	7.096
λ_{kd}	0.1	4.888	6.633
λ_{kd}	**0.5**	**4.712**	**6.367**
λ_{kd}	1	4.879	6.499
λ_{kd}	2	4.935	6.731

4.6. Dynamic Gate Weight Analysis

To further investigate whether the proposed routing mechanism can adapt to different scene densities and spatial layouts, we visualize several paired examples with their corresponding gate weights in Figure 6. Each pair contains two scenes from the same dataset or category but with different object density, spatial distribution, or background complexity. The results show that the routing weights are not fixed across images; instead, they vary according to the visual structure of the scene. This observation supports the claim that SAGRD performs input-dependent scale routing rather than static multi-scale fusion.

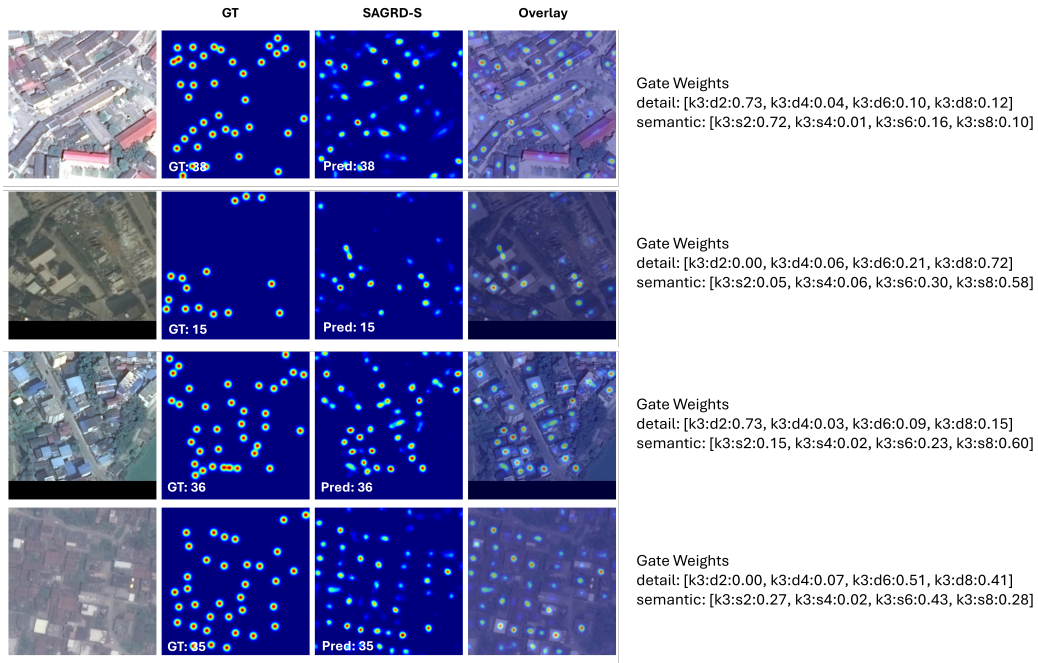


Figure 6. Visualization of the proposed dynamic gate mechanism on the RSOC dataset. From left to right, the columns show the original images, ground-truth density maps, predicted density maps, and prediction overlays. In the density maps, brighter cyan/yellow responses indicate higher density values, while dark blue indicates low or near-zero density. The gate weights vary with object distribution and scene complexity, illustrating content-dependent soft routing across receptive-field branches.

The results show that the learned gate weights vary noticeably across different scenes, indicating that SAGRD does not use a fixed multi-scale fusion pattern. For Scene 835, which contains relatively dense and complex building structures, both the detail and semantic routers assign dominant weights to the smallest receptive-field branch, with 0.73 and 0.72 for the k3-d2 branch, respectively. This suggests that the model relies more on local fine-grained responses when targets are compact and spatial details are important.

In contrast, Scene 685 has fewer targets and a more sparse spatial distribution. The gate weights shift strongly toward the largest receptive-field branch. The detail router assigns 0.72 to k3-d8, and the semantic router assigns 0.58 to k3-d8. This indicates that the model adaptively increases the use of larger receptive fields when broader contextual information is needed for sparse or dispersed target regions.

Scene 846 and Scene 721 further demonstrate that the routing behavior is not determined only by the object count. Although both scenes have similar counts, their gate distributions are different. In Scene 846, the detail branch still focuses on k3-d2, while the semantic branch assigns a high weight to k3-d8. This implies that the model combines local detail extraction with broader semantic context. In Scene 721, the detail router mainly activates middle-to-large receptive fields, with high weights on k3-d6 and k3-d8, while the semantic router distributes weights across k3-d2, k3-d6, and k3-d8. This more balanced routing pattern reflects the different spatial layout and background structure of the scene.

Overall, these visualizations confirm the dynamic nature of the proposed gate mechanism. The routing weights change according to scene complexity, target density, spatial distribution, and background structure, rather than following a fixed branch preference. Therefore, SAGRD can adaptively emphasize different receptive-field branches across remote sensing scenes, supporting the formulation of multi-scale fusion as content-dependent soft routing rather than static aggregation.

4.7. Efficiency and Deployment Analysis

Importantly, SAGRD uses soft routing and evaluates all K branches; therefore, its parameter and FLOP reductions are mainly provided by the compact backbone, reduced channel widths, depthwise-separable expert/refinement blocks, and lightweight context/fusion modules rather than by skipping low-weight branches. The first frame is excluded when computing the average latency because it includes warm-up overhead such as CUDA initialization, memory allocation, and kernel loading.

Table 14 reports the embedded-device inference performance on the Jetson Orin Nano platform under different input resolutions. Overall, the proposed SAGRD-Student achieves substantially lower computational complexity than the teacher model while maintaining a clear runtime advantage at medium and high resolutions. Specifically, the student reduces FLOPs from 14.754 G to 1.296 G at 256 × 256 input resolution and from 59.015 G to 5.184 G at 512 × 512, corresponding to approximately 11.4× theoretical computation reduction. Meanwhile, the measured latency decreases from 98.39 ms to 68.56 ms at 256 × 256 and from 174.90 ms to 98.31 ms at 512 × 512, achieving 1.44× and 1.78× practical speedup, respectively. These results indicate that the lightweight student becomes increasingly advantageous as the input resolution grows.

However, the latency reduction is noticeably smaller than the FLOPs reduction, especially at low resolutions. For example, at 128 × 128 input size, the student reduces FLOPs from 3.689 G to 0.324 G but only improves latency from 65.57 ms to 64.01 ms. This phenomenon suggests that the deployment bottleneck on embedded GPUs is not determined solely by arithmetic computation. Instead, lightweight routing-based networks are often constrained by memory access, kernel-launch overhead, and operator scheduling efficiency. In particular, operations such as deformable convolution, dynamic gating, tensor reshaping, Softmax routing, and depthwise convolution introduce relatively low FLOPs but non-negligible runtime overhead on embedded hardware.

Compared with extremely lightweight methods such as EdgeCount, the proposed SAGRD-Student still has higher latency despite maintaining competitive efficiency. This is mainly because SAGRD contains dynamic routing and adaptive receptive-field operations, which are computationally efficient but less hardware-friendly than purely static

convolutional architectures. Nevertheless, SAGRD consistently achieves a more favorable balance between counting accuracy and deployment efficiency than conventional heavy-counting models such as ASPDNet and PSGCNet, whose latency increases rapidly under high-resolution inputs. Therefore, the experimental results demonstrate that the proposed framework is suitable for practical embedded deployment, while also revealing the remaining hardware overhead introduced by adaptive routing mechanisms.

Table 14. Comparison of inference latency and FPS on NVIDIA Jetson Orin Nano.

Methods	Params	Input Size	FLOPs	Orin Nano	
				Latency (ms)	FPS
ASPDNet [2]	23.03 M	128	9.70 G	44.40	22.52
		256	38.79 G	99.69	10.03
		512	155.16 G	353.71	2.83
PSGCNet [19]	27.51 M	128	8.26 G	20.23	19.44
		256	32.35 G	44.25	22.60
		512	128.69 G	171.50	5.83
LMSFFNet [21]	4.58 M	128	117.76 M	44.47	22.48
		256	476.55 M	48.15	20.77
		512	1.99 G	83.73	11.94
LMSFFNet-S [21]	9.51 M	128	868.49 M	39.35	25.42
		256	3.47 G	44.85	22.30
		512	13.89 G	91.33	10.95
EdgeCount [22]	0.12 M	128	19.87 M	40.46	24.72
		256	84.75 M	41.90	23.87
		512	423.91 M	66.45	15.05
SAGRD-T	14.19 M	128	3.689 G	65.57	15.42
		256	14.754 G	98.39	10.37
		512	59.015 G	174.90	5.91
SAGRD-S	1.39 M	128	0.324 G	64.01	16.37
		256	1.296 G	68.56	15.25
		512	5.184 G	98.31	10.39

4.8. Limitation Analysis

Despite its effectiveness, SAGRD still has several limitations. First, the performance remains sensitive to extremely dense scenes. When the target density is much higher than that of the training distribution, the density map may become locally saturated, resulting in undercounting. This issue is particularly relevant for small-vehicle scenes, where many instances are densely aligned and partially visually merged.

Second, background structures with object-like geometry may still produce false responses. In remote sensing images, roofs, road markings, shadows, docks, and repetitive textures can resemble target objects in local patterns. Although the scale-aware router can suppress some background interference by selecting more appropriate receptive fields, it does not explicitly model object-background semantic uncertainty.

Third, as formulated in Equation (19), the current routing distillation operates on image-level routing distributions obtained by spatially averaging the dense gate maps. This design intentionally avoids over-constraining the lightweight student with per-pixel supervision, yet it also means that intra-image spatial variations in the teacher's routing decisions are collapsed into a single probability vector. Consequently, the student is encouraged to match the teacher's global scale-selection behavior for the entire scene, but it is not required to reproduce where the teacher assigns smaller or larger receptive

fields within specific local regions. Future work may explore spatially explicit routing distillation—for example by aligning low-resolution gate maps or distilling object-region-specific routing statistics—to transfer finer-grained spatial routing knowledge. However, such objectives must be designed carefully, because enforcing overly strong spatial gate matching may over-constrain the student and reduce the flexibility that is critical for lightweight optimization.

Fourth, the current model is evaluated under a dataset-wise supervised setting. This setting is valid for verifying counting performance when each dataset has separated training and test splits. Nevertheless, cross-dataset generalization remains an important future direction. Since remote sensing data vary greatly in platform, spatial resolution, target category, and annotation style, future work should further test whether the learned routing behavior transfers across domains when no target-domain labels are available.

Fifth, the current study uses a frozen teacher under an offline distillation protocol and does not determine whether joint teacher–student optimization could further raise the student performance ceiling. In addition, annotation conversion and high-resolution resizing introduce local supervision uncertainty: geometric reference points may not exactly coincide with visual object centroids, while downsampling can suppress fine details and reduce the apparent scale of tiny targets. Future work will investigate controlled joint optimization together with native-resolution tiling or resolution-adaptive training to address these limitations without excessive memory cost.

5. Conclusions

This work presents Scale-Aware Gated Routing Distillation (SAGRD), a lightweight remote sensing object counting framework designed to preserve scale-adaptive counting ability under limited computational resources.Different from conventional lightweight counting methods that mainly reduce backbone complexity or distill only final density maps, SAGRD formulates multi-scale feature fusion as a dynamic routing problem and transfers the teacher model's scale-selection behavior to the student model.

The main experimental results show that SAGRD achieves an effective balance between counting accuracy and model efficiency. On the RSOC Building evaluation setting, the teacher model obtains an MAE/RMSE of 4.678/6.329 with 14.186 M parameters and 14.83 G FLOPs, while the student model achieves a close MAE/RMSE of 4.712/6.367 with only 1.395 M parameters and 1.29 G FLOPs. This corresponds to approximately 90.2% parameter reduction and 91.3% FLOPs reduction compared with the teacher model. The ablation studies further verify the effectiveness of the proposed components, including scale-aware routing, branch response feedback, spatial routing logits, density-map distillation, and gate-routing distillation. These results indicate that learning the teacher's routing behavior is beneficial for lightweight remote sensing counting, especially when the student model has limited representational capacity.

Author Contributions: Conceptualization, Z.Z.; methodology, Z.Z. and J.Z.; software, Z.Z. and J.Z.; validation, Z.Z. and J.Z.; formal analysis, Z.Z. and J.Z.; investigation, J.Z.; data curation, Z.Z. and J.Z.; hardware deployment and embedded-platform testing, Y.G.; writing—original draft preparation, Z.Z. and J.Z.; writing—review and editing, H.M.; supervision, H.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used in this study are publicly available benchmark datasets, as described and cited in the experimental section of the manuscript. No new private dataset was generated in this study.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hsieh, M.R.; Lin, Y.L.; Hsu, W.H. Drone-based object counting by spatially regularized regional proposal network. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4165–4173.
2. Gao, G.; Liu, Q.; Wang, Y. Counting from sky: A large-scale data set for remote sensing object counting and a benchmark method. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 3642–3655. [CrossRef]
3. Liu, W.; Salzmann, M.; Fua, P. Context-aware crowd counting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5099–5108.
4. Ma, Z.; Wei, X.; Hong, X.; Gong, Y. Bayesian loss for crowd count estimation with point supervision. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6142–6151.
5. Ma, Z.; Wei, X.; Hong, X.; Lin, H.; Qiu, Y.; Gong, Y. Learning to count via unbalanced optimal transport. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 2319–2327.
6. Li, X.; Wang, W.; Hu, X.; Yang, J. Selective kernel networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 510–519.
7. Chen, Y.; Dai, X.; Liu, M.; Chen, D.; Yuan, L.; Liu, Z. Dynamic convolution: Attention over convolution kernels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11030–11039.
8. Yang, B.; Bender, G.; Le, Q.V.; Ngiam, J. CondConv: Conditionally parameterized convolutions for efficient inference. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
9. Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
10. Fedus, W.; Zoph, B.; Shazeer, N. Switch Transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.* **2022**, *23*, 5232–5270.
11. Yang, Y.; Jiang, P.-T.; Hou, Q.; Zhang, H.; Chen, J.; Li, B. Multi-task dense prediction via mixture of low-rank experts. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 17–21 June 2024; pp. 27927–27937.
12. Pavlitska, S.; Fan, H.; Ditschuneit, K.; Zöllner, J.M. Design and behavior of sparse mixture-of-experts layers in CNN-based semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Denver, CO, USA, 3–7 June 2026; pp. 4272–4281.
13. Lempitsky, V.; Zisserman, A. Learning to count objects in images. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 6–9 December 2010; pp. 1324–1332.
14. Zhang, Y.; Zhou, D.; Chen, S.; Gao, S.; Ma, Y. Single-image crowd counting via multi-column convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 589–597.
15. Li, Y.; Zhang, X.; Chen, D. CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1091–1100.
16. de Almeida, P.R.L.; Oliveira, L.S.; Britto, A.S., Jr.; Silva, E.J., Jr.; Koerich, A.L. PKLot—A robust dataset for parking lot classification. *Expert Syst. Appl.* **2015**, *42*, 4937–4949. [CrossRef]
17. Gao, J.; Gong, M.; Li, X. Global multi-scale information fusion for multi-class object counting in remote sensing images. *Remote Sens.* **2022**, *14*, 4026. [CrossRef]
18. Guo, X.; Anisetti, M.; Gao, M.; Jeon, G. Object counting in remote sensing via triple attention and scale-aware network. *Remote Sens.* **2022**, *14*, 6363. [CrossRef]
19. Gao, G.; Liu, Q.; Hu, Z.; Li, L.; Wen, Q.; Wang, Y. PSGCNet: A pyramidal scale and global context guided network for dense object counting in remote-sensing images. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5619412. [CrossRef]
20. Zhu, X.; Hu, H.; Lin, S.; Dai, J. Deformable ConvNets V2: More deformable, better results. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9308–9316.
21. Yi, J.; Shen, Z.; Chen, F.; Zhao, Y.; Xiao, S.; Zhou, W. A lightweight multiscale feature fusion network for remote sensing object counting. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5902113. [CrossRef]
22. Shen, Z.; Li, G.; Xia, R.; Meng, H.; Huang, Z. A lightweight object counting network based on density map knowledge distillation. *IEEE Trans. Circuits Syst. Video Technol.* **2025**, *35*, 1492–1505. [CrossRef]
23. Duan, Z.; Wang, S.; Di, H.; Deng, J. Distillation remote sensing object counting via multi-scale context feature aggregation. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5613012. [CrossRef]
24. Huang, Y.; Jin, Y.; Zhang, L.; Liu, Y. Remote sensing object counting through regression ensembles and learning to rank. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 4701617. [CrossRef]

25. Zhu, L.; Zhao, Z.; Lu, C.; Lin, Y.; Peng, Y.; Yao, T. Dual path multi-scale fusion networks with attention for crowd counting. *arXiv* **2019**, arXiv:1902.01115.
26. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
27. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
28. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 116–131.
29. Mehta, S.; Rastegari, M. MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer. In Proceedings of the International Conference on Learning Representations, Virtual, 25–29 April 2022.
30. Wang, P.; Gao, C.; Wang, Y.; Li, H.; Gao, Y. MobileCount: An efficient encoder–decoder framework for real-time crowd counting. *Neurocomputing* **2020**, *407*, 292–299. [CrossRef]
31. Liu, Y.; Cao, G.; Shi, H.; Hu, Y. Lw-Count: An effective lightweight encoding-decoding crowd counting network. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 6821–6834. [CrossRef]
32. Yi, J.; Chen, F.; Shen, Z.; Xiang, Y.; Xiao, S.; Zhou, W. An effective lightweight crowd counting method based on an encoder–decoder network for the Internet of Video Things. *IEEE Internet Things J.* **2024**, *11*, 3082–3094. [CrossRef]
33. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
34. Liu, L.; Chen, J.; Wu, H.; Chen, T.; Li, G.; Lin, L. Efficient crowd counting via structured knowledge transfer. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 2645–2654.
35. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
36. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
37. Wu, Y.; He, K. Group normalization. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 3–19.
38. Li, J.; Wen, Y.; He, L. SCConv: Spatial and channel reconstruction convolution for feature redundancy. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 6153–6162.
39. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
40. Stahl, T.; Pintea, S.L.; van Gemert, J.C. Divide and count: Generic object counting by image divisions. *IEEE Trans. Image Process.* **2019**, *28*, 1035–1044. [CrossRef]
41. Sindagi, V.A.; Patel, V.M. Generating high-quality crowd density maps using contextual pyramid CNNs. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1861–1870.
42. Sam, D.B.; Surya, S.; Babu, R.V. Switching convolutional neural network for crowd counting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4031–4039.
43. Meng, Y.; Zhang, H.; Zhao, Y.; Yang, X.; Qian, X.; Huang, X.; Zheng, Y. Spatial uncertainty-aware semi-supervised crowd counting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 15549–15559.
44. Wang, C.; Song, Q.; Zhang, B.; Wang, Y.; Tai, Y.; Hu, X.; Wang, C.; Li, J.; Ma, J.; Wu, Y. Uniformity in heterogeneity: Diving deep into count interval partition for crowd counting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 3234–3242.
45. Song, Q.; Wang, C.; Jiang, Z.; Wang, Y.; Tai, Y.; Wang, C.; Li, J.; Huang, F.; Wu, Y. Rethinking counting and localization in crowds: A purely point-based framework. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 3365–3374.
46. Chen, B.; Yan, Z.; Li, K.; Li, P.; Wang, B.; Zuo, W.; Zhang, L. Variational attention: Propagating domain-specific knowledge for multi-domain learning in crowd counting. *Int. J. Comput. Vis.* **2022**, *130*, 2293–2314.
47. Sindagi, V.A.; Patel, V.M. CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance, Lecce, Italy, 29 August–1 September 2017; pp. 1–6.
48. Shen, Z.; Xu, Y.; Ni, B.; Wang, M.; Hu, J.; Yang, X. Crowd counting via adversarial cross-scale consistency pursuit. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5245–5254.
49. Sam, D.B.; Babu, R.V. Top-down feedback for crowd counting convolutional neural network. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 7323–7330.

50. Cao, X.; Wang, Z.; Zhao, Y.; Su, F. Scale aggregation network for accurate and efficient crowd counting. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 734–750.
51. Gao, J.; Wang, Q.; Li, X. PCC Net: Perspective crowd counting via spatial convolutional network. *arXiv* **2019**, arXiv:1905.10085.
52. Ma, X.; Du, S.; Liu, Y. A lightweight neural network for crowd analysis of images with congested scenes. In Proceedings of the IEEE International Conference on Image Processing, Taipei, Taiwan, 22–25 September 2019; pp. 979–983.
53. Shi, X.; Li, X.; Wu, C.; Kong, S.; Yang, J.; He, L. A real-time deep network for crowd counting. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Barcelona, Spain, 4–8 May 2020; pp. 2328–2332.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.