SJER Scientific Journal of Engineering Research

futuristech PT. Teknologi Futuristik Indonesia

Received: April 13, 2026; Revised: May 23, 2026; Accepted: May 25, 2026; Published: May 29, 2026; Current version: May 29, 2026. e-ISSN: 3109-1725
Digital Object Identifier 10.64539/sjer.v2i3.2026.493

Article

A Study of Loss Weight Balance in Lightweight Self-Distilled Crowd Counting

Muhammad Raza[1,*], Atta Ur Rahman[2], Pandula Pallewatta[3], Inayat Ur Rahman[2], Sahib Bahadar[4]

[1] School of Computer Science, Nanjing University of Information Science & Technology, Nanjing, 210044, China; e-mail: mrazabng125@gmail.com (M. Raza).
[2] School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing, 210044, China.
[3] School of Artificial Intelligence, Nanjing University of Information Science & Technology, Nanjing, 210044, China.
[4] School of Electronics and Communication Engineering, Nanjing University of Information Science & Technology, Nanjing, 210044, China.
* Correspondence Author

The authors received no financial support for the research, authorship, and/or publication of this article.

Abstract: Lightweight crowd counting is important for real-time surveillance and resource-constrained deployment, where both computational efficiency and effective supervision are required. Although teacher-free self-distillation can improve lightweight density-regression models by guiding intermediate representations without an external teacher, the influence of composite loss weights in such frameworks has not been sufficiently analyzed. This paper presents a focused coefficient-wise loss-weight analysis within the Lightweight Self-Knowledge Distillation framework for single-image crowd counting. Instead of proposing a new architecture, the study investigates how the coefficients α, β, γ, and λ_2 affect optimization behavior and counting accuracy under a fixed experimental setup on ShanghaiTech Part B. Specifically, α controls intermediate feature alignment, β controls consistency supervision, γ controls direct density-regression supervision, and λ_2 controls the structural similarity term in the regression loss. The results show that moderate values of α and β improve performance by providing useful internal regularization, while excessive auxiliary weighting can slightly degrade accuracy. The analysis also indicates that γ should remain dominant because direct density-map regression is the primary learning signal. The best observed configuration is α = 6.0, β = 2.0, γ = 13.0, and λ_2 = 0.2, achieving 8.94 MAE and 11.51 RMSE on ShanghaiTech Part B. These findings highlight the importance of balanced supervision design within the evaluated LSKD framework on ShanghaiTech Part B.

Keywords: Crowd counting; Lightweight crowd counting; Self-knowledge distillation; Composite loss weighting; Density map regression.

1. Introduction

Crowd counting is a fundamental task in computer vision with important applications in public safety surveillance, intelligent transportation, event monitoring, and urban management [1]. In real-world contexts, crowd counting systems need to approximate the number of individuals in extremely dense scenes where extreme occlusion, scale change, background clutter, and perspective distortion render localization to be inaccurate [2]. This is why density map regression has taken the paradigm of current-day crowd counting, as it is capable of maintaining the information of spatial distribution whilst the resulting count can be determined by integrating the density map that the method predicts [3]. Although significant progress has been made in deep learning, it is still challenging to estimate crowds in real-world settings with high accuracy, as thick crowd images cannot be analyzed with either high local or global contextual information [4].

Meanwhile, the constraints on model complexity are very high at the time of practical deployment. Most of the high-performing crowd counting algorithms are based on deep backbones, multi-scale branches, or computationally intensive modules, which are challenging to implement in real-time and resource-limited settings. This has resulted

in an increasing interest in lightweight models of counting crowds [5] that minimize parameters and floating-point computations but retain competitive accuracy [6]. Lightweight architectures are associated with a trade-off; however, they are more efficient, but have a lower representational capacity, which may make it harder to support stable learning of multi-level features and robust density estimation in dense scenes [7]. This means that lightweight crowd counting cannot be based purely on architectural simplification, but must also have useful supervision strategies that enhance internal representations without adding to the cost of inference-time.

One approach that can be used to deal with this challenge is knowledge distillation [8]. Compact models are enhanced by traditional teacher-student distillation by copying information on a large external teacher into a small network of students. Such methods work well in a variety of environments, but because they add complexity to training and are not always appropriate in the dense regression problems like crowd counting, where the continuity of space and the regularity of structure are particularly valuable. Teacher-free self-distillation offers a more effective solution [9], as it allows transfer of knowledge within a single model [10]. This concept is applied in the Lightweight Self-Knowledge Distillation (LSKD) [11] framework introduced above by using internal feature alignment, and contextual enhancement, with a lightweight backbone and a Feature Matching Block (FMB), a Context Fusion (CoFuse) module, and a combined training goal that includes intermediate alignment loss, consistency loss, and direct density-regression loss.

The published LSKD [11] article defined the framework as an effective lightweight crowd counting solution, whereas the thesis also explored its ablation behaviour and parameter sensitivity. Nevertheless, the behavior of the composite loss weighting is a question that is not well-discussed independently. In objectives of multi-term self-distillation, the performance at the end does not only rely on the availability of auxiliary supervision, but also the strengths of the weighting of each term as part of the optimization. When auxiliary losses are too small, they do not give significant regularization; when they are too large, they can over-regularize the learning of features and make it less flexible to the actual counting goal. Such a problem is of special concern when counting crowds with a lightweight model, where smaller models may be advantageous due to the ability to add more supervision, but where they are also more vulnerable to unreliable optimization.

This gap motivates a focused coefficient-wise analysis of loss-weight balancing in lightweight self-distilled crowd counting. Instead of introducing a new architecture, the paper explores the influence of weighting parameters in the LSKD objective on the optimization behavior and counting accuracy with a fixed training configuration. In particular, we examine the sensitivity of the intermediate alignment coefficient α, the consistency coefficient β, the regression coefficient γ and the structural similarity coefficient $\lambda 2$. We evaluate them on ShanghaiTech Part B with Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). This study helps analyze the balance between auxiliary supervision and direct density regression within the LSKD framework under the ShanghaiTech Part B setting by isolating the effect of these variables.

The contributions of this paper are summarized as follows. First, this work presents a focused analytical study of the composite loss weights used in the existing LSKD framework. Second, it examines the coefficient-wise effects of α, β, γ and $\lambda 2$ on counting accuracy under a controlled one-factor-at-a-time experimental setting on ShanghaiTech Part B. Third, it reports the best observed configuration within the evaluated coefficient sweep and discusses the relative roles of intermediate alignment, consistency supervision, direct density regression, and structural similarity in the LSKD objective. This work does not introduce a new architecture, module, or training objective; rather, it provides an empirical analysis of how individual loss coefficients influence the behavior of the existing LSKD framework under the evaluated ShanghaiTech Part B setting.

The remainder of this paper is organized as follows. Section 2 reviews related work on crowd counting, lightweight crowd-counting models, knowledge distillation, and self-knowledge distillation. Section 3 presents the LSKD objective and describes the loss coefficients analyzed in this study. Section 4 explains the experimental setup, dataset, evaluation metrics, and coefficient-wise sensitivity-analysis protocol. Section 5 reports and discusses the effects of α, β, γ, and λ_2 on counting performance. Section 6 and Section 7 present the main discussion, limitations, and future research directions. Finally, Section 8 concludes the paper.

2. Related Work

2.1. Crowd Counting

Crowd counting has undergone 3 major paradigms namely: detection-based methods, regression-based methods and density map estimation. The early detection-based methodologies tried to locate individual persons directly, however, their analysis performance seriously declined in a crowded scene due to extreme occlusion, overlap, and cluttering of the background [12]. Regression-based algorithms minimized the use of explicit localization, but instead learned a mapping between image characteristics to the total count, but still relied heavily on handcrafted representations and could offer limited spatial clues [13]. Density map regression has become the formulation of choice with the emergence of deep learning,

as it computes a spatially interpretable density surface the integral of which gives the final crowd count. The given formulation works especially well with congested scenes due to the fact that it maintains spatial structure without directly separating instances.

The latest models of crowd counting have enhanced density estimation by using multi-scale feature extraction [14], contextual reasoning, encoder-decoder [15], dilated convolution and attention [16]. With these developments, it is now possible to deal with variations in scale, distortion of perspective, and unclear background patterns in a better way. Simultaneously, they also demonstrate that effective crowd counting performance relies not only on network depth or feature richness [17], but also on the effectiveness with which a model integrates local detail and the higher-level semantic and contextual information [18].

2.2. Lightweight Crowd Counting

Whereas the recent density-regression approaches have reached an impressive accuracy, most of the most successful crowd counting models are computationally costly and inapplicable in practice in real-time or under resource constraints. The constraint has sparked a growing interest in lightweight crowd counting networks that minimize parameters and floating-point operations at the cost of competitive performance. Current lightweight methods have delved into the exploration of efficient backbones [19] like MobileNetV2 and GhostNet, lightweight multi-scale modules, compact encoder-decoder designs [20], channel attention and dynamic convolution.

These works demonstrate that there can be large compression in computation [21], but they also reveal a common constraint: when dealing with dense and highly cluttered scenes, compact architectures can be difficult to support any robust contextual reasoning and consistent multi-level features. That is, scaling models down is not the sole solution to the issue, as lightweight models might lose representational richness at the point of crowd counting most challenging. It is one of the primary reasons why lightweight models frequently need more powerful supervision strategies along with the simplification of architecture.

2.3. Knowledge Distillation for Crowd Counting

Knowledge distillation is another popular approach to enhancing smaller neural networks by transferring knowledge contained in a more powerful teacher model to a smaller student. This concept is particularly appealing to crowd counting, as lightweight models can tend to be more sensitive to richer supervision to maintain semantic distinctions and spatial coherence in density estimation. Previous research has investigated feature-level distillation, attention transfer, structured knowledge transfer, and hybrid schemes that reconcile between local and global supervision [22]. These experiments demonstrate that distillation has the potential to enhance counting accuracy and generalization especially in dense regression environments where small models would otherwise fail to capture the structure of complex scenes.

Nevertheless, there are also significant practical and methodological constraints of teacher-student distillation. It needs an extra teacher network, complicates training, and can bias or density-map artifacts onto the student. Such concerns are particularly pertinent in terms of crowd counting, where spatial consistency is a factor, and lightweight deployment is a significant goal in many cases. Consequently, despite the fact that compact models can be enhanced with teacher-student distillation, it is not necessarily the most effective or stable method of lightweight density estimation.

2.4. Self-Knowledge Distillation for Crowd Counting

Self-knowledge distillation offers a teacher-free solution where supervision is passed in-network across layers or stages [23]. In contrast to traditional distillation, self-distillation does not need an external teacher, and hence it minimizes the training overhead, but still enables deeper representations to teach shallow ones. Previous research has established that self-distillation can enhance generalization, stabilize optimization, and enhance the quality of features without making inference-time more complex. The property renders it especially applicable to dense prediction tasks, such as crowd counting, where more semantic and contextual information is captured in deeper layers and the more detailed spatial information is captured in earlier layers.

Internal feature consistency can be used in crowd counting to enable the network to generate more consistent and spatially coherent density maps. It is also proposed in the literature that self-distillation is more effective when applied together with contextual modeling [24] as ambiguous local patterns in crowded scenes can usually be resolved with global information about a scene. Simultaneously, self-distillation is not free of difficulties: intermediate and deep features can vary in channel dimension, semantics, and spatial resolution and naive alignment can disrupt learning. This is why the key to self-distillation in dense regression lies not just in feature adaptation, but also in a loss design that trades internal alignment with the main density-estimation goal [25].

2.5. Research Gap

In spite of the fact that previous research has led to lightweight crowd counting and self-distillation, it has mostly focused on architecture, efficiency, and final benchmark outcomes but not the optimization behavior of composite training objectives. In models like LSKD, the optimization of the intermediate alignment, consistency su-

pervision, and direct density regression is done together and the performance is highly dependent on the weighting of these terms with each other. Weak auxiliary weighting may provide insufficient regularization, whereas excessive weighting may over-constrain feature learning and reduce counting fidelity. Although this problem is significant, the weight distribution of composite loss functions in teacher-free lightweight crowd counting is not studied sufficiently. This paper addresses this gap through a focused coefficient-wise analysis of the LSKD loss coefficients under a fixed experimental setting. The analysis is conducted as a one-factor-at-a-time parameter sweep, where one coefficient is varied while the remaining coefficients and training configuration are kept unchanged.

3. LSKD Objective and Analysis Scope

The framework used in the current study is LSKD [11] for single-image crowd counting, which was proposed earlier. In the original architecture, a lightweight density-regression network is combined with teacher-free self-distillation to enhance feature learning without using an external teacher model. The backbone first extracts hierarchical feature representations from the input image. The Feature Matching Block (FMB) then transforms intermediate feature maps so that they become more compatible with the deeper guidance feature used for self-distillation. This step is necessary because intermediate and deep features may differ in channel dimension, semantic level, and spatial representation, making direct alignment less stable. After feature adaptation, the Context Fusion (CoFuse) module further refines the transformed intermediate features by incorporating contextual information from deeper representations. As a result, the features used for intermediate supervision contain both local spatial details and stronger semantic guidance. Finally, the regression head generates the predicted density map from the deeper representation. These components are the primary architectural contribution of the original LSKD paper. In the present paper, however, the framework serves only as the analytical basis for studying the composite training objective and its weighting coefficients. Therefore, FMB and CoFuse are briefly described here because they construct the transformed feature used in the intermediate alignment loss, but they are not treated as new contributions of this study.

3.1. Minimal Framework Context

Given an input image I, the lightweight backbone produces a set of intermediate feature maps $\{H_i\}_{i=1}^{N}$ and a deeper semantic feature map H_d. The final density prediction D_p is obtained by a regression head from deeper representation and the final crowd count is the accumulation of all the values in the predicted density map. In training, self-distillation is used to make deeper representations guide shallower and intermediate layers to enable the network to enhance internal semantic consistency without adding higher inference-time complexity.

Since intermediate and deep features vary in terms of channel dimension and semantic level, FMB initially adapts every intermediate feature after which CoFuse refines it before applying supervision. These modules play a significant role in the initial LSKD framework since they specify the feature paths in terms of internal supervision. In the present study, however, they are not treated as new contributions or independent objects of analysis. Their role here is only to provide the feature representations on which the loss terms operate. The main question of this paper is therefore not how the architecture is built, but how the existing supervision terms should be balanced during optimization.

3.2. Composite Loss Formulation

The training goal for LSKD includes the combination of three types of supervision, which are intermediate feature alignment, deep-feature consistency, and direct density-regression supervision [11]. All of these losses are optimized simultaneously, which implies that the ultimate performance of the system is influenced by the weight given to each loss type.

First, after feature adaptation and context enhancement, each intermediate feature representation is aligned with the deeper semantic feature. Let H_i denote the i-th intermediate feature map, $\widetilde{H}_i$ its aligned version after FMB, and H_i^c its context-enhanced version after CoFuse. The intermediate supervision is defined through a feature-alignment loss:

$$L_{IA}^{(i)} = \frac{1}{BCHW} |H_i^c - H_d|_F^2 \tag{1}$$

where B is the batch size, and C, H, and W denote the channel, height, and width of the aligned feature representation. The notation $|\cdot|_F^2$ represents the squared Frobenius norm. Therefore, $L_{IA}^{(i)}$ is computed as the element-wise mean squared difference between the transformed intermediate feature and the deep guidance feature and the total intermediate alignment loss is obtained by averaging across all supervised intermediate stages:

$$L_{\mathrm{IA}} = \frac{1}{N} \sum_{i=1}^{N} L_{\mathrm{IA}}^{(i)} \tag{2}$$

This term acts as an internal teacher-free distillation signal, encouraging intermediate layers to learn semantically stronger representations under the guidance of deeper features.

Second, a consistency loss is introduced to strengthen the relationship between the deep semantic feature and

the final density prediction D_p. In the original LSKD formulation, the deep feature is transformed into a density-like representation through channel-wise averaging and then compared with the predicted density map:

$$L_{cons} = \frac{1}{BHW}|Avg(H_d) - D_p|_F^2 \quad (3)$$

where $Avg(\cdot)$ denotes channel-wise averaging of the deep feature H_d. This operation converts the multi-channel deep feature into a single-channel density-like representation. B, H, and W denote the batch size, height, and width of the resulting map respectively. The notation $|\cdot|_F^2$ represents the squared Frobenius norm. Therefore, L_{cons} is computed as the element-wise mean squared difference between the averaged deep feature representation and the predicted density map. Since the channel dimension is removed by $Avg(\cdot)$, the normalization term is BHW, not BCHW. This loss encourages the final density prediction to remain consistent with the semantic structure captured by the deep feature representation.

Third, direct supervision is applied to the predicted density map through a regression loss that combines pixel-wise numerical fidelity and structural similarity:

$$L_{\text{reg}} = \lambda_1|D_p - D|^2 + \lambda_2\left(1 - \text{SSIM}\big(D_p, D\big)\right) \quad (4)$$

where D is the ground-truth density map, $\lambda 1$ controls the numerical regression term, and $\lambda 2$ controls the contribution of structural similarity. In the original training setup, $\lambda 1$ is fixed to 1.0, while $\lambda 2$ is treated as a supporting coefficient rather than a dominant one.

The overall training objective is therefore written as:

$$L_{\text{total}} = \alpha L_{\text{IA}} + \beta L_{\text{cons}} + \gamma L_{\text{reg}} \quad (5)$$

where α, β, and γ determine the relative importance of intermediate alignment, consistency supervision, and direct regression, respectively. In this paper, the coefficient $\lambda 2$ is also analyzed because it governs the internal balance of the regression objective itself. The four parameters considered in the sensitivity study therefore play distinct roles:

- α controls the strength of intermediate self-distillation through feature alignment;
- β controls the consistency constraint between deep features and predicted density;
- γ controls the dominance of direct density regression in the total objective;
- $\lambda 2$ controls the contribution of structural similarity within the regression loss.

This formulation makes the LSKD objective more expressive than a standard density-regression loss, but it also makes optimization more sensitive. Performance depends not only on whether all of the supervision terms are present, but also on whether their relative strengths match the actual hierarchy of the crowd-counting task.

3.3. Analytical Motivation

The main idea of this study is that, within the evaluated LSKD framework, auxiliary supervision should act as a controlled regularizer rather than as a competing objective. If α is too small, intermediate alignment may provide limited benefit; if it is too large, the network may be overly constrained to follow deeper representations. A similar trade-off exists for β. Weak consistency supervision may not help stabilize things very much, while too much consistency pressure may make it harder to reach the final counting goal. The role of γ is different because direct density regression should remain the primary learning signal in the crowd-counting objective. Lastly, $\lambda 2$ determines how much structural similarity should affect the regression objective compared to pixel-wise density matching.

Therefore, these coefficients should not be treated as interchangeable hyperparameters. Each one controls a different part of supervision, and good optimization depends on keeping the right balance between them. The goal of this study is to identify the observed operating region where this balance improves performance within the evaluated LSKD framework on ShanghaiTech Part B.

3.4. Scope of the present Analysis

The contribution of this paper is analytical rather than architectural or methodological. The study does not introduce new modules, replace the backbone, propose a new loss function, or aim to establish a new state-of-the-art result across multiple datasets. Instead, it uses the existing LSKD framework as a fixed experimental platform to examine the coefficient-wise behavior of the composite loss objective under a fixed training setting on ShanghaiTech Part B. The reported best setting should therefore be interpreted as the best observed configuration within the current one-factor-at-a-time coefficient sweep, rather than as a universally optimal setting or a new method. This scope allows the effect of each loss coefficient to be examined individually while keeping the architecture and training protocol unchanged. Because the analysis is conducted on the authors' previously proposed LSKD framework, independent validation on other frameworks is required before drawing broader conclusions.

4. Experimental Design

The present work evaluates the influence of composite loss coefficients within a fixed LSKD training configuration. To ensure transparency, reproducibility, and controlled comparison, all major experimental settings are kept unchanged across the sensitivity experiments. The

Table 1. Summary of the ShanghaiTech Part B dataset used in the sensitivity study.

Dataset	Images (Train/Test)	Annotations	Count Range	Avg Count
ShanghaiTech Part B	716 (400/316)	~90,000	9–578	123

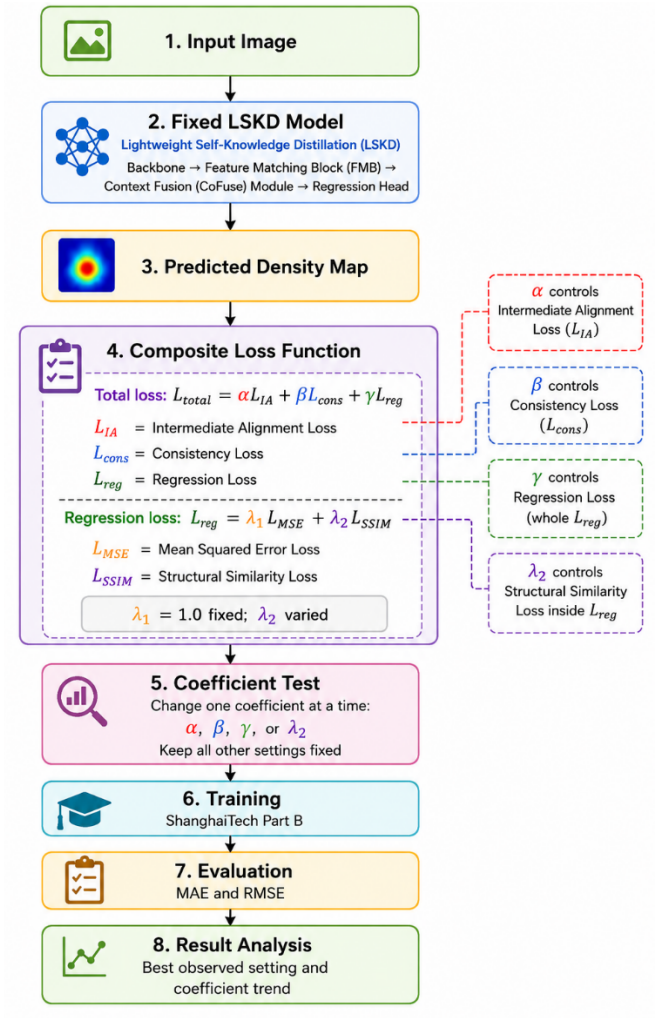

Figure 1. Methodological workflow of the coefficient-wise loss-weight analysis in the fixed LSKD framework. One coefficient is varied at a time, while the remaining settings are kept fixed, and each configuration is evaluated on ShanghaiTech Part B using MAE and RMSE.

experiments are conducted using the PyTorch framework on the ShanghaiTech Part B dataset, which contains 400 training images and 316 testing images. During training, input images are randomly cropped to 512×512 and horizontally flipped for data augmentation. Ground-truth density maps are generated using a fixed Gaussian kernel with a kernel size of 15. The model is optimized using the Adam optimizer for 150 epochs with a batch size of 2 and an initial learning rate of 1×10^{-4}. The evaluation is performed using MAE and RMSE. All experiments are conducted on an NVIDIA GeForce RTX 4080 GPU. For the reference configuration, the loss coefficients are set as α=6.0, β=2.0, γ=13.0, $\lambda1$=1.0 and $\lambda2$=0.2. In each sensitivity analysis, only one target coefficient is varied while the remaining coefficients and all other training settings are fixed. To make the methodology clearer, the overall research stage, fixed LSKD framework, and coefficient-wise testing strategy are summarized in Figure 1. The dataset and analysis setting are as follows.

4.1. Dataset and Analysis Settings

ShanghaiTech Part B, introduced by Zhang et al. [3], is used as the controlled benchmark in this study. The dataset has 716 street-scene images, as indicated in Table 1, and a 400/316 train-test split and almost 90,000 annotations. ShanghaiTech Part B offers a medium-density environment, in which the effect of loss coefficients can be more evident, compared to more irregular and highly congested datasets. With an average crowd of 123 and a range of counts between 9 and 578, it provides a relatively stable setting for examining coefficient-wise performance trends. In this paper, ShanghaiTech Part B is used not to claim a new state-of-the-art result, but to provide a controlled setting in which the effect of each loss coefficient can be examined.

4.2. Training and Configuration

The same training configuration is maintained across all experiments. Training images are augmented using random cropping and horizontal flipping, following the original LSKD configuration [11]. For ShanghaiTech Part B, the cropped image size is 512×512, and the ground-truth density maps are generated using a fixed Gaussian kernel with a kernel size of 15. The model is implemented in PyTorch and optimized using Adam for 150 epochs with a batch size of 2. All experiments are conducted on an NVIDIA GeForce RTX 4080 GPU. The initial learning rate is set to 1×10^{-4} and is reduced in later training stages to support convergence. The reference operating point is based on the original LSKD setup; α=6.0, β=2.0, γ=13.0, $\lambda1$=1.0, and $\lambda2$=0.2. In each coefficient-wise analysis, only one parameter is varied while all other coefficients and training settings are kept fixed to isolate the effect of the selected coefficient. Each coefficient configuration is trained and evaluated once under the same training protocol; therefore, the reported MAE and RMSE values are single-run results rather than averages over multiple independent experiments. Random seeds and explicit deterministic CUDA settings are not enforced in the current study; therefore, minor stochastic variation may remain due to random initialization, data augmentation, and GPU-level operations. This limitation is considered when interpreting the results as best observed single-run values rather than averages over repeated independent trials.

4.3. Evaluation Metrics

The evaluation of performance is carried out by Mean Absolute Error (MAE), Root Mean Square Error (RMSE) which are conventional measures of performance in crowd counting. Here, N is the number of test images and $D_i^{pred}(x,y)$ and $D_i^{gt}(x,y)$ are the predicted and ground-truth values of the density at location *(x, y)* in the *i*-th image. The measures are calculated in the following way:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\sum_{x,y} D_i^{pred}(x,y) - \sum_{x,y} D_i^{gt}(x,y)\right| \quad (6)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{x,y} D_i^{pred}(x,y) - \sum_{x,y} D_i^{gt}(x,y)\right)^2} \quad (7)$$

MAE quantifies the mean counting error but RMSE is more severe on large deviations and is employed to test robustness and stability.

4.4. Sensitivity Analysis Protocol

The sensitivity analysis follows a one-factor-at-a-time protocol. In each experiment, only one coefficient is varied while the remaining coefficients are fixed at the reference setting. First, α is varied to examine the influence of intermediate feature alignment. Second, β is varied to evaluate the effect of consistency supervision. Third, γ is varied to study the role of direct density-regression supervision. Finally, $\lambda 2$ is varied while $\lambda 1$ is fixed to observe the contribution of the structural similarity term. This protocol does not aim to perform exhaustive hyperparameter optimization; rather, it is designed to isolate the individual effect of each coefficient within the LSKD objective.

4.5. Controlled Comparison Principle

In each experiment, only one coefficient is modified, while the backbone, feature modules, crop size, optimizer, learning schedule, training epochs, and evaluation protocol are kept constant. This design helps ensure that the observed trends are mainly due to loss-weight sensitivity rather than architectural or procedural differences.

4.6. Purpose of the Experimental Design

The experimental design is intended to answer a focused question: what balance of auxiliary and direct supervision yields stable optimization in the evaluated LSKD framework on ShanghaiTech Part B? The next section presents the coefficient-wise results for α, β, γ, and $\lambda 2$, and interprets what these trends reveal about effective loss balancing in LSKD.

5. Coefficient-wise Analysis of Loss Weights

This section examines how the composite loss coefficients affect crowd-counting performance in the LSKD framework under the fixed ShanghaiTech Part B experimental setting. Since α, β, γ and $\lambda 2$ represent different aspects of the training objective, their effects are discussed separately to clarify their respective contributions to optimization. Since each coefficient setting was evaluated without repeated independent runs, the reported MAE and RMSE values represent single-run results within the current controlled coefficient-wise sweep, rather than averages over multiple random seeds.

5.1. Overall Best Setting

The best observed results in this controlled setting are achieved at α=6.0, β=2.0, γ=13.0, and $\lambda 2$=0.2, yielding 8.94 MAE and 11.51 RMSE on ShanghaiTech Part B. This configuration is used as the reference point for the coefficient-wise analysis. The observed trend suggests that the objective performs better when auxiliary supervision is strong enough to regularize learning, while the regression term remains dominant. The changing values of these coefficients can be visualized in Figure 2.

5.2. Effect of α

The sensitivity analysis of α is given in Table 2. When β=2.0 and γ=13.0 are fixed, the best observed performance is obtained at α=6.0. In particular, MAE decreases from **9.43** to **8.94**, and RMSE decreases from **12.18** to **11.51**. When α is increased further, there is a slight decrease in performance, to MAE 9.05 and RMSE 11.66 at 8.0 and MAE 9.28 and RMSE 11.94 at 10.0.

These findings indicate that intermediate feature alignment is helpful when it provides sufficient semantic guidance to shallower layers. However, excessive alignment may over-constrain intermediate features and reduce their flexibility in supporting the main density-regression task. Therefore, α should be treated as a regularization coefficient rather than a dominant supervision factor.

5.3. Effect of β

Table 3 shows sensitivity analysis of β. With α=6.0 and γ=13.0 fixed, β=0.5 gives 9.24 MAE and 11.93 RMSE, while β=1.0 improves performance to MAE 9.08 and RMSE 11.71. The best observed result in this sweep is obtained when β=2.0, where the model achieves 8.94 MAE and 11.51 RMSE. When β is increased further, performance decreases slightly to MAE 9.01 and RMSE 11.61 at β=3.0 and MAE 9.16 and RMSE 11.79 at β=4.0.

This trend indicates that consistency supervision is helpful when used in a moderate range. A small value offers a little regularization, but a large value causes the auxiliary consistency constraint to be too restrictive. The best observed value of β is therefore obtained when consistency supervision enhances training stability without dominating the overall loss.

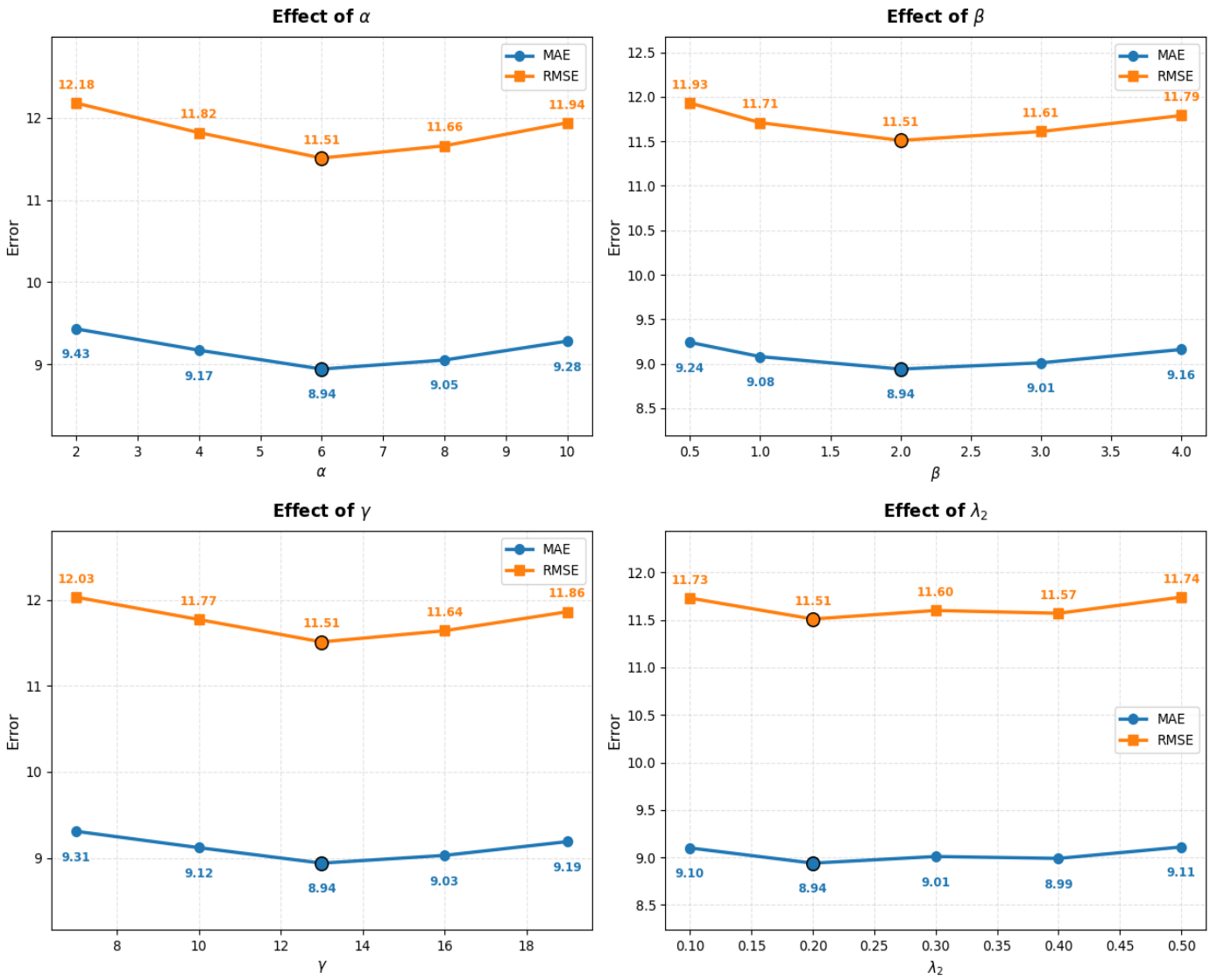

Figure 2. Sensitivity analysis of loss coefficients on ShanghaiTech Part B.

Table 2. Sensitivity analysis of α on ShanghaiTech Part B.

α	β	γ	MAE	RMSE
2.0	2.0	13.0	9.43	12.18
4.0	2.0	13.0	9.17	11.82
6.0	2.0	13.0	**8.94**	**11.51**
8.0	2.0	13.0	9.05	11.66
10.0	2.0	13.0	9.28	11.94

Table 3. Sensitivity analysis of β on ShanghaiTech Part B.

α	β	γ	MAE	RMSE
6.0	0.5	13.0	9.24	11.93
6.0	1.0	13.0	9.08	11.71
6.0	2.0	13.0	**8.94**	**11.51**
6.0	3.0	13.0	9.01	11.61
6.0	4.0	13.0	9.16	11.79

5.4. Effect of γ

The sensitivity analysis of γ is reported in Table 4. When α=6.0 and β=2.0 are kept constant, the performance improves as γ increases from 7.0 to 13.0 to give the best observed result of 8.94 MAE and 11.51 RMSE at γ=13.0. When γ is increased further, performance again declines slightly, with MAE 9.03 and RMSE 11.64 at γ=16.0 and MAE 9.19 and RMSE 11.86 at γ=19.0.

The coefficient γ, unlike α and β, regulates the main regression goal. This trend suggests that direct density regression should remain the prevailing learning signal in the evaluated setting. When γ is too small, the model receives insufficient direct task supervision; when it is too large, the contribution of auxiliary self-distillation terms becomes less effective.

5.5. Effect of λ2

Table 5 shows the sensitivity analysis of λ2. With λ1=1.0 fixed, performance improves from MAE 9.10 and RMSE 11.73 at λ2=0.1 to the best values MAE 8.94 and RMSE 11.51 at λ2=0.2. When λ2 is increased further, performance decreases slightly to MAE 9.01 and RMSE 11.60 at λ2=0.3, MAE 8.99 and RMSE 11.57 at λ2=0.4 and MAE 9.11 and RMSE 11.74 at λ2=0.5.

Table 4. Sensitivity analysis of γ on ShanghaiTech Part B.

α	β	γ	MAE	RMSE
6.0	2.0	7.0	9.31	12.03
6.0	2.0	10.0	9.12	11.77
6.0	2.0	13.0	**8.94**	**11.51**
6.0	2.0	16.0	9.03	11.64
6.0	2.0	19.0	9.19	11.86

Table 5. Sensitivity analysis of λ2 on ShanghaiTech Part B.

λ1	λ2	MAE	RMSE
1.0	0.1	9.10	11.73
1.0	0.2	**8.94**	**11.51**
1.0	0.3	9.01	**11.60**
1.0	0.4	8.99	11.57
1.0	0.5	9.11	11.74

These findings indicate that the structural similarity term is useful when it acts as a supporting refinement term within the regression objective. However, it should remain secondary to the pixel-wise regression component. Over-emphasis on structural similarity may shift optimization away from numerical counting fidelity.

5.6. Joint Interpretation

Overall, the coefficient-wise results show that the LSKD objective is sensitive to the relative weighting of its supervision terms. Moderate values of α and β improve the observed performance, while larger values reduce accuracy, indicating that auxiliary supervision is useful only when it remains controlled. The results for γ show that direct density regression should retain the largest contribution within the objective. The λ2 sweep further indicates that structural similarity is helpful as a supporting term but should not dominate the pixel-wise regression component. The best observed setting in the current controlled sweep, α=6.0, β=2.0, γ=13.0 and λ2=0.2, reflects this balance.

6. Discussion

6.1. Supervision Balance in LSKD

The results suggest that the composite LSKD objective benefits from a clear hierarchy among its loss terms. In the evaluated setting, auxiliary feature alignment and consistency supervision improve performance when they support the main density-regression task, but excessive weighting can shift optimization away from the final counting objective. This behavior is particularly relevant for lightweight models, where additional supervision can help compensate for limited representation capacity but may also over-constrain feature learning. Therefore, the observed trend supports the use of auxiliary losses as controlled regularizers rather than competing optimization targets.

6.2. Implications for the Evaluated LSKD Setting

In the evaluated LSKD setting on ShanghaiTech Part B, the results indicate that supervision design remains important even when the network architecture is fixed. The findings suggest that lightweight self-distilled models may benefit from internal supervision, but the contribution of each loss term should be controlled according to its role in the objective. This supports the view that improving performance in the evaluated LSKD framework is not only a matter of architectural efficiency, but also of appropriately balancing the training signals used to guide the model.

6.3. Computational Overhead and Training Dynamics

Changing the loss coefficients α, β, γ and λ2 does not modify the LSKD network architecture, backbone, FMB, CoFuse module, or regression head. Therefore, the coefficient changes do not increase the number of model parameters or inference-time computational cost. However, different loss weights can affect the training process by changing the relative contribution of each loss term to the total optimization objective. For example, larger auxiliary weights may increase the influence of intermediate alignment or consistency supervision, which can affect gradient balance, training stability, and convergence behavior. Similarly, increasing γ strengthens the direct density-regression term and may guide the model more strongly toward the final counting objective.

In the current study, the analysis focuses on final MAE and RMSE values rather than detailed training dynamics. Therefore, convergence speed, training curves, gradient magnitudes, and epoch-wise stability are not explicitly analyzed. Future work will include training-loss curves, convergence comparisons, and gradient-level analysis to better understand how loss-weight changes affect optimization behavior.

6.4. Practical Implication

In the evaluated LSKD framework, loss-weight selection should not be treated as a minor implementation detail because it affects final counting performance under the current ShanghaiTech Part B setting. The observed results suggest that direct density regression should remain the primary supervision signal, while feature alignment, consistency supervision, and structural similarity should be used as supporting terms. This provides a practical reference for selecting loss weights when applying the existing LSKD framework under similar experimental conditions.

6.5. Comparison with previous studies

Previous crowd-counting studies have improved performance through different strategies. Multi-column methods such as MCNN address scale variation, but their accuracy is limited compared with later models. Deeper

density-regression methods such as CSRNet improve performance in congested scenes, but they require higher computational cost. Lightweight models such as MobileCount, LSANet, and Lw-Count reduce parameters and FLOPs for efficient deployment, but compact networks may still have limited representation capacity in complex crowd scenes. Teacher-student distillation methods can improve small models, but they require an external teacher during training and may introduce teacher-dependent bias. In contrast, the LSKD framework uses teacher-free internal supervision. Unlike these previous studies, the present work does not propose a new architecture; instead, it analyzes how the loss weights in the existing LSKD objective affect counting performance. Therefore, its main strength is the focused analysis of supervision balance, while its limitation is that the study is restricted to LSKD on ShanghaiTech Part B.

7. Limitations and Future Work

This study has several limitations. First, the sensitivity analysis is conducted only on the ShanghaiTech Part B dataset. Although this dataset provides a stable and commonly used benchmark for crowd-counting evaluation, the best observed setting, α=6.0, β=2.0, γ=13.0 and $\lambda 2$=0.2 should not be assumed to be universally optimal for all datasets. Crowd-counting datasets differ in crowd density, image resolution, perspective distortion, annotation distribution, and scene complexity, which may affect the relative importance of each loss component. Therefore, the conclusions of this study are limited to the LSKD framework under the ShanghaiTech Part B experimental setting. Future work will extend the analysis to additional datasets such as ShanghaiTech Part A, UCF-QNRF, and UCF-CC-50.

Second, the current study uses a one-factor-at-a-time sensitivity analysis, where one coefficient is varied while the remaining coefficients are fixed. This design helps isolate the effect of each loss coefficient, but it does not fully capture possible interactions among α, β, γ, and $\lambda 2$. Future studies may use a broader grid search or factorial experimental design to analyze joint coefficient interactions more comprehensively.

Third, the reported results are based on the fixed LSKD framework proposed in previous work. Therefore, the conclusions are specific to this framework and may not directly generalize to other lightweight crowd-counting architectures or self-distillation methods. Future research should validate whether similar loss-weight trends appear in other teacher-free or lightweight density-regression models.

Fourth, this study is based on the authors' previously proposed LSKD framework [11]. Since the same internal framework serves as the experimental platform for all coefficient-wise analyses, the findings may be influenced by framework-specific design choices, including the lightweight backbone, FMB, CoFuse, and the original composite loss structure. Therefore, the observed loss-weight trends should be interpreted as specific to the evaluated LSKD setting rather than as independently validated conclusions for all lightweight self-distilled crowd-counting models. Future work should involve independent validation by applying similar coefficient-wise analyses to other lightweight crowd-counting architectures and teacher-free self-distillation frameworks.

Finally, the present work mainly focuses on final counting accuracy measured by MAE and RMSE. The reported results are not based on repeated independent trials, confidence intervals, or statistical significance testing. Therefore, small numerical differences, such as 8.94 MAE versus 9.01 MAE, should be interpreted cautiously. In this paper, the term "best observed" refers to the lowest MAE and RMSE obtained within the current controlled coefficient-wise sweep, rather than a statistically verified optimum. Future work will include repeated trials with different random seeds, standard deviation reporting, confidence intervals, and convergence analysis to provide stronger statistical and optimization-level evidence. In addition, this study does not explicitly report convergence speed, training-loss curves, gradient magnitudes, or training-time changes caused by different loss-weight settings.

8. Conclusion

This paper presented a focused coefficient-wise analysis of the composite loss weights in the existing LSKD framework under the ShanghaiTech Part B setting. Instead of introducing a new architecture or training objective, the study examined how α, β, γ, and $\lambda 2$ affect counting performance under a fixed experimental protocol. The best observed configuration in the current controlled sweep is α=6.0, β=2.0, γ=13.0, and $\lambda 2$=0.2, yielding 8.94 MAE and 11.51 RMSE. The results suggest that, within the evaluated LSKD setting, the composite objective benefits from balanced loss weighting, where auxiliary self-distillation terms support but do not dominate the main density-regression supervision. This study complements the original LSKD method paper by providing an empirical analysis of loss-weight behavior. Future work will validate these observations using additional datasets, repeated random seeds, and independent lightweight self-distillation frameworks.

9. Declarations

9.1. Author Contributions

Muhammad Raza: Conceptualization, Methodology, Software, Data Curation, Formal Analysis, Investigation, Visualization, Writing – Original Draft, **Atta Ur Rahman:** Formal Analysis, Investigation, Validation, Writing – Review & Editing; **Pandula Pallewatta:** Review and Editing; **Inayat ur Rahman:** Investigation, Validation; **Sahib Bahadar:** Writing, Review and Editing.

9.2. Institutional Review Board Statement

Not applicable.

9.3. Informed Consent Statement

Not applicable.

9.4. Data Availability Statement

The dataset used in this study is the publicly available ShanghaiTech Part B benchmark dataset introduced by Zhang et al. [3]. The dataset is cited through its original publication.

9.5. Acknowledgment

Not applicable.

9.6. Conflicts of Interest

The authors declare no conflict of interest.

10. References

[1] V. A. Sindagi and V. M. Patel, "A survey of recent advances in CNN-based single image crowd counting and density estimation," *Pattern Recognition Letters*, vol. 107, pp. 3–16, 2018. https://doi.org/10.1016/j.patrec.2017.07.007.

[2] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2013, pp. 2547–2554. https://doi.org/10.1109/CVPR.2013.329.

[3] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 589–597. https://doi.org/10.1109/CVPR.2016.70.

[4] Y. Li, X. Zhang, and D. Chen, "CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 1091–1100. https://doi.org/10.1109/CVPR.2018.00120.

[5] A. G. Howard *et al*., "MobileNets: Efficient convolutional neural networks for mobile vision applications,"*arXiv preprint* arXiv:1704.04861, 2017. https://doi.org/10.48550/arXiv.1704.04861.

[6] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 4510–4520. https://doi.org/10.1109/CVPR.2018.00474.

[7] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 1580–1589. https://doi.org/10.1109/CVPR42600.2020.00165.

[8] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015. https://doi.org/10.48550/arXiv.1503.02531.

[9] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 4320–4328. https://doi.org/10.1109/CVPR.2018.00454.

[10] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma, "Be your own teacher: Improve the performance of convolutional neural networks via self distillation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 3713–3722. https://doi.org/10.1109/ICCV.2019.00381.

[11] M. Raza, M. Ling, A. U. Rahman, P. Pallewatta, A. A. Hersi, S. M. Beruwalage, and D. S. Kannangara, "LSKD: Lightweight self-knowledge distillation framework for fast and robust crowd counting," *Scientific Journal of Engineering Research*, vol. 2, no. 2, 2026. https://doi.org/10.64539/sjer.v2i2.2026.436.

[12] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2008, pp. 1–7. https://doi.org/10.1109/CVPR.2008.4587569.

[13] V. Lempitsky and A. Zisserman, "Learning to count objects in images," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2010, pp. 1324–1332. https://proceedings.neurips.cc/paper_files/paper/2010/file/fe73f687e5bc5280214e0486b273a5f9-Paper.pdf.

[14] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 5744–5752. https://doi.org/10.1109/CVPR.2017.429.

[15] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale aggregation network for accurate and efficient crowd counting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 757–773. https://openaccess.thecvf.com/content_ECCV_2018/papers/Xinkun_Cao_Scale_Aggregation_Network_ECCV_2018_paper.pdf.

[16] X. Jiang, Z. Xiao, B. Zhang, X. Zhen, X. Cao, D. Doermann, and L. Shao, "Crowd counting and density estimation by trellis encoder-decoder networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 6133–6142. https://doi.org/10.1109/CVPR.2019.00629.

[17] Z. Ma, X. Wei, X. Hong, and Y. Gong, "Bayesian loss for crowd count estimation with point supervision," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 6142–6151. https://doi.org/10.1109/ICCV.2019.00624.

[18] W. Liu, M. Salzmann, and P. Fua, "Context-aware crowd counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 5099–5108. https://doi.org/10.1109/CVPR.2019.00524.

[19] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 116–131. https://openaccess.thecvf.com/content_ECCV_2018/papers/Ningning_Light-weight_CNN_Architecture_ECCV_2018_paper.pdf.

[20] C. Gao, P. Wang, and Y. Gao, "MobileCount: An efficient encoder-decoder framework for real-time crowd counting," in *Pattern Recognition and Computer Vision: Second Chinese Conference*, 2019, pp. 582–595. https://doi.org/10.1007/978-3-030-31723-2_50.

[21] Y.-B. Liu, G. Cao, H. Shi, and Y. Hu, "Lw-Count: An effective lightweight encoding-decoding crowd counting network," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 32, no. 10, pp. 6821–6834, 2022. https://doi.org/10.1109/TCSVT.2022.3171235.

[22] T. Furlanello, Z. C. Lipton, M. Tschannen, L. Itti, and A. Anandkumar, "Born-again neural networks," in *Proc. 35th Int. Conf. Mach. Learn. (ICML)*, 2018, pp. 1602–1611. https://proceedings.mlr.press/v80/furlanello18a/furlanello18a.pdf.

[23] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 1195–1204. https://proceedings.neurips.cc/paper_files/paper/2017/file/68053af2923e00204c3ca7c6a3150cf7-Paper.pdf.

[24] V. A. Sindagi and V. M. Patel, "HA-CCN: Hierarchical attention-based crowd counting network," *IEEE Trans. Image Process.*, vol. 29, pp. 323–335, 2020. https://doi.org/10.1109/TIP.2019.2928634.

[25] B. Wang, H. Liu, D. Samaras, and M. Hoai, "Distribution matching for crowd counting," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020, vol. 33, pp. 1595–1607. https://proceedings.neurips.cc/paper_files/paper/2020/file/118bd558033a1016fcc82560c65cca5f-Paper.pdf.