Scientific Journal of Engineering Research

futuristech
PT. Teknologi Futuristik Indonesia

Date of publication March 2, 2026, date of current version March 2, 2026.
e-ISSN: 3109-1725
Digital Object Identifier 10.64539/sjer.v2i2.2026.436

Article

LSKD: Lightweight Self-Knowledge Distillation Framework for Fast and Robust Crowd Counting

Muhammad Raza[1,*], Miaogen Ling[1], Atta Ur Rahman[2], Pandula Pallewatta[3], Aboubakar Abdinur Hersi[1], Shehan Maxwell Beruwalage[1], Deshan Sachintha Kannangara[1]

[1] School of Computer Science, Nanjing University of Information Science & Technology, Nanjing, 210044, China; mrazabng125@gmail.com
[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
* Correspondence

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

Abstract: Crowd counting plays an important role in the surveillance of the safety of the people, traffic, and intelligent surveillance systems. However, the exact density estimations remain hard to achieve in highly congested scenes due to the tough occlusion, large-scale variance, and complicated background. Although the recent deep-learning methods have high performance, several of them do not need computationally efficient underlying backbone networks, and rather, they employ an external teacher-student distillation architecture, which can limit their use in resource-constrained applications. To avoid this problem, we introduce LSKD, a lightweight self-knowledge distillation network that is density map regression-specific. Unlike other conventional teacher-dependent processes, LSKD can also independently carry out internal multi-level feature alignment within a single small network that is not in need of an external teacher model. The structure integrates a Feature Matching Block (FMB) and a Context Fusion (CoFuse) block to enhance the hierarchical match of features and global awareness of context. The large experiments demonstrate that LSKD obtain competitive performance using the number of parameters as 2.65 million and GFLOPs as 10.23. Particularly, it has 63.17 MAE on ShanghaiTech Part A, 8.94 on ShanghaiTech Part B, 143.7 on UCF-QNRF, and 223.88 on UCF-CC-50, which is a good ratio between the accuracy and the efficiency of the calculations. Such results indicate that LSKD has an implementable and efficient solution to the real-time counting of crowds at the edge devices.

Keywords: Crowd counting; Teacher–student distillation; Self-knowledge distillation; Density map regression; Lightweight network; Multi-level feature alignment.

1. Introduction

In this study, we define lightweight model as an architecture that is specified to operate under limited computational budgets, typically containing fewer than 5 million parameters, and require fewer than 15 GFLOPs. They can normally be used on edge devices or other embedded, low processing-capability platforms. In this sense, efficient or low cost-of-computation models are networks capable of competing in crowd-counting performance on a much smaller parameter size and much smaller number of floating-point operations than the traditional deep networks modeling crowd-counting, which may have tens of millions of parameters, and have much higher computational complexity.

Crowd counting is a key task in computer vision that estimates the number of people in images or videos and has serious applications in public safety surveillance, traffic congestion, urban management, and even intelligent transportation systems. Although there have been significant breakthroughs in deep learning, precise crowd counting has not been achieved in real-world conditions due to intense occlusion, high perspective distortion, large-scale variation, background clutter, and extremely uneven crowd distributions [1]. The challenges are compounded in dense scenes, where heavy overlap among individuals creates local visual ambiguity and unstable density estimates, especially under real-time, resource-limited deployment constraints [2].

These challenges are exacerbated in large gatherings where people are packed together, and the line of sight is poor. The two main paradigms of early crowd counting practices were detection-based counting and regression-based counting on handcrafted features. Detection-based methods represent each individual using localized density distributions and aggregate counts. Still, they often fail in crowded scenes where severe occlusion and heavy overlap make it difficult to distinguish instances [3]. The only way to avoid this is by density map estimation, which does not focus on localization but instead produces a density map, resulting in poor inter-scene generalization [4]. This formulation maintains space information and is more efficient in highly congested spaces [5].

With the success of deep learning, density regression based on CNNs has become the standard approach, as it offers strong representation learning. The change in perspective and viewpoint is one of the challenges of crowd scenes, as it alters scale. This is done using multi-scale architectures, such as multi-column networks, which derive features from receptive fields of different scales to learn crowd patterns at various scales [6]. Along with fixed multi-scale designs, there are dynamically weighted predictors based on local density, applied via adaptive routing and switching mechanisms to increase robustness in non-uniform scenes [7]. The feature aggregation based on context also improves density estimation by exploiting context and multi-level semantic features to reduce confusion between the crowd and background areas that resemble the crowd. Hybrid detection-regression models combine the abilities of detection in low-density regions and density regression in high-density areas to better fulfill the large intra-image density variations [8]. The developments indicate that successful crowd counting requires excellent multi-scale representations, along with appropriate reasoning in the context. The combination of these architectural enhancements suggests that, to accurately count a crowd, there must be a delicate balance between the representation of multi-scale features and feature-context reasoning to handle extreme density variations.

Other than in architecture, the stability and reliability of the density estimations are largely reliant on the supervision design and optimization targets. The overlap of crowds and confusion in annotations may make pure pixel-wise regression less consistent, due to background congestion. The idea of composition loss was proposed to jointly promote correct counting, high-density estimation, and localization enhancement, thereby providing more information about the supervision signals [9]. Bayesian loss model annotation uncertainty, and the model is more robust with noisy point supervision [10]. The distribution-matching strategies are associated with the predicted and target distributions, resulting in improved density map quality and better training in a highly congested environment [11]. These results indicate that the combination of the backbone design with good performance in crowd counting, as well as goals that consist of the spatial and statistical characteristics of density maps, yields good results. Nevertheless, some of these acquired supervision methods are often associated with complex constructions and are difficult to incorporate into lightweight crowd arrangement models directly.

Although newer ones are said to be very precise, many of these are either heavy backbones or complex modules that increase computational and memory costs, thereby reducing their usability in real-time applications on edge devices [12]. This sparks the introduction of light systems that do not shortchange on efficiency. MobileNet, ShuffleNetV2, and GhostNet are among the best backbones that reduce FLOPs and parameters by leveraging depth-wise separable convolution, implementation-aware design, and cheap feature generation [13]. However, counting crowds with the simple application of compact networks remains non-trivial due to the need for strong multi-level feature fusion and workable semantic context-functionalities, which cannot be sustained in lightweight frameworks under the harsh computational constraints [14].

Among the trending methods for improving the capacity of small models is knowledge distillation (KD), in which knowledge acquired by a large-capacity model is shared with a smaller-capacity student model [15]. The decision to direct the intermediate representations is also demonstrated through feature-level distillation and attention transfer, which are particularly determined by spatially structured tasks [16]. However, KD is computationally expensive and tends to introduce teacher heterogeneity or density-map artifacts, which can be especially detrimental in dense regression problems where error is exponential across the predicted density distribution [17]. This paper is a mere adaptation of the self-knowledge distillation to the density map regression; in contrast to the other existing self-distillation algorithms, which are geared towards self-presenting classification or recognition, in the present case, it is essential to preserve the spatial layout of the dense crowd scenes by the consistency of hierarchical feature presentations.

Under these restrictions, teacher-free means of knowledge transfer that retain the advantages of distillation while not relying on a decent teacher are inspired. In particular, this type of internal knowledge transfer is appropriate for density-based crowd counting, as cross-feature-hierarchy consistency can be applied to foster spatial consistency and improve count stability, without increasing inference cost. Self-distillation is an informative regularizer that can convey credible information across layers of training or across training phases, but is not grounded in an autonomous network of teachers [18]. Deep Mutual Learning, Born-Again Neural Networks, and the mean teacher demonstrate that self-guided, consistent learning can enhance generalization and stability without an external teacher [19]. Specifically, they appeal to the lightweight crowd-counting technique, in which accuracy, robustness, and efficiency must be balanced [20].

Even though a lot of work has been done in terms of crowd counting, the trade-off between accuracy and computation efficiency has not been addressed. Models that perform well are typically based on deep backbones, multi-scale sophisticated modules, or external teacher-student distillation structures that make them more complex to train and expensive to deploy. On the other hand, lightweight models minimize parameters and FLOPs and can occasionally fail to retain multi-level feature persistence and resilient global contextual decision-making in dense and crowded scenes. Additionally, the current knowledge distillation algorithms used in crowd counting largely rely on the presence of external teacher networks that bring about extra computational cost and bias that may occur due to the teachers.

Although both lightweight architectures and distillation-based learning have been shown to be effective on their own, little has been done to combine them into one, unified and teacher-free system designed specifically to run density map regression. Specifically, the existing methods of self-knowledge distillation seldom aim to maintain (spatial) coherence and structural consistency, which are essential to dense crowd estimation. To fill this gap, our paper introduces LSKD, a simple self-knowledge distillation model, which allows internal feature alignment and global contextual consistency even in a single network, which is not based on an external teacher. The aim is to work towards an even-handed solution that guarantees accuracy of counting, structural resilience, and computational efficiency to be deployed in resource-constrained environments.

Building on the above observations, the paper proposes a lightweight architecture for crowd counting, designed for real time deployment and enhanced feature learning via knowledge transfer. The significant contributions in this work are the following:

- We develop a light architecture to count crowds with the assistance of density map regression, which fits within conditions of severe computational limits.
- We present a self-knowledge distillation mechanism that, in the absence of an external teacher model, allows aligning internal features without the assistance of a teacher.
- We suggest a Feature Matching Block (FMB) and a Context Fusion (CoFuse) module to increase the multi-level feature consistency and global contextual understanding.

The rest of the paper is structured as follows. Section 2 reviews related work on crowd counting, lightweight architectures, and knowledge distillation and self-knowledge distillation. Section 3 presents the proposed lightweight self-knowledge distillation framework in detail. Experimental settings and quantitative results are reported in Section 4, followed by ablation studies in Section 5. Finally, Section 6 concludes the paper and discusses potential future directions.

2. Background and Related Work

2.1. Crowd Counting

Counting people in an image or video, typically to determine their spatial distribution, is called crowd counting. This application is used in other areas, such as public safety, traffic control, and event management. This is because manual counting methods have been extensively researched, as large or dense scenes cannot be counted manually. Its initial strategies were based on manual elements and hints of hybridity. Idrees et al. used low-confidence head detection, frequency analysis, and texture statistics in a multi-source, multi-scale system, which is also prone to occlusion and clutter, yet still managed to show that it can partially handle a very crowded scene [2]. Topkaya et al. applied a person detector and a Dirichlet Process Mixture Model to cluster noisy outputs and estimate the number of people without prior knowledge of the number of targets, even though the detector's output was heavily affected by its implementation and handcrafted representations [21]. The development of deep learning spurred research on CNN-based density map regression.

Zhang et al. introduced MCNN, a multi-column CNN with varying receptive fields to address large head-scale variation, and geometry-adaptive Gaussian kernels to produce density maps, which represent a major step towards replacing handcrafted features with end-to-end deep models [6]. The latter development focused on more effective fusion backbones and multi-scale fusion. The CSRNet substitutes the multi-column architecture with a single VGG-16 front-end and a dilated convolutional back-end, achieving high accuracy in crowded scenes at the expense of a comparatively heavy network [22]. Chen et al. streamlined this concept in SPN by using a single deep backbone and a Scale Pyramid Module built on parallel dilated convolutions, which are more effective at capturing multi-scale context and more accurate at counting [23]. To further improve the quality of density maps, Jiang et al. proposed TEDnet, a trellis encoder-decoder with trellis-encoder-decoding paths and dense skip connections; they train the network on a combinatorial loss that enforces both local coherence and spatial correlation, resulting in even better-quality density maps and more visually realistic results [24].

SFANet provides attention to this line of work via an attention map path and a density map path over a VGG16 backbone, jointly optimized by attention and density losses, which ensure the network focuses on the head regions and prevents background clutter [25]. In more recent work, several studies have focused on computational efficiency and deployment. Shi et al. proposed a small real-time CNN with an architecture based on a simple multi-scale front layer, followed by a single-branch density estimator, achieving an adequate trade-off between accuracy and speed for real-time tasks [26]. In the aerial platform case, Chen et al. proposed Flounder-Net, which employed interleaved group convolutions and aggressive spatial down-sampling to operate on high-resolution drone imagery on embedded hardware, achieving accuracy comparable to that of heavier fully convolutional networks with many fewer parameters and significantly faster inference [27]. In addition to human crowds, PSGCNet introduces a pyramidal scale module and a global context module, along with an enhanced Bayesian counting loss for dense object counting on complex remote-sensing images. The proposed model also projects crowd images with perfect accuracy [28].

Generally, advances in crowd counting technologies have been trending towards handcrafted pipelines, detection pipelines, and deep-density-regression pipelines with multi-scale and attention mechanisms. In recent times, several studies have focused on the trade-off between accuracy and computational efficiency to enable real-time deployment. The past models that had good performance were more likely to have a deep backbone and numerous complex multi-scale modules; however, the recently developed lightweight models demonstrate that it is possible to achieve the required performance on models with significantly reduced parameters and FLOPs. Nevertheless, high multi-level feature consistency and contextual reasoning are hard to maintain in small designs, particularly in highly dense and highly cluttered scenes. Therefore, it is very encouraging to explore more deployment-friendly models that would not affect the efficiency and structural stability.

2.2. Light Weight Networks

Lightweight crowd-counting designs are challenging due to the high demand for fine-grained spatial information, high-scale modeling, and rich semantic context, all of which are typically compromised when model capacity and computational budget are severely constrained. A lightweight crowd-counting study aims to minimize model size without compromising the quality of the multi-scale representation. Guo et al. introduced GAPNet, which combines a GhostNet backbone, a zero-parameter channel attention mechanism, and a compact pyramid fusion module to address scale variation at low computational cost [29]. Li et al. proposed a lightweight, dense estimation network comprising an L-weight module with pointwise, depthwise, and stacked convolutions, along with pyramid aggregation and channel-attention fusion to improve multi-scale feature representation while keeping the network small [30]. Zhu et al. proposed LSANet, a real-time model with hybrid dilated convolutions to extract multi-scales and an efficient attention-based fusion to achieve higher MIoU and better distinction between foreground and crowd regions [31]. Jiang et al. introduced LigMSANet, which builds upon a customized backbone of MobileNetV2 and a multi-scale adaptive module that dynamically changes receptive fields to adapt to crowd size at low parameter counts [32]. Wang et al. developed a lightweight MobileNetV2 + dilated convolution network for edge nodes, achieving faster inference at a very low accuracy cost through density estimation on real-time IoT devices [33].

Chaudhuri et al. introduced the resource-efficient ASFNet, which is based on MobileNet or MobileViT backbones and adjacent feature fusion to combine multi-scale context with significantly lower FLOPs and parameters [34]. The lightweight designs can be broadly grouped into GhostNet-based schemes that leverage cheap feature generation, MobileNet-based schemes that use depthwise separable convolutions, or dynamic/curriculum-based schemes that are less adaptive but much smaller. The more recent lightweight crowd-counting models aim to improve efficiency through greater compactness and better feature extraction. Chen et al. introduced a lightweight hybrid model that uses GhostNet to generate local features and a Swin-Transformer with a modified global context to perform count prediction with less supervision and computational cost [35].

Li et al. proposed a dynamic convolutional network (DCN) that uses MobilenetV2, in which dynamic kernels and curriculum reinforcement learning enhance versatility and training consistency, without increasing model size [36]. The Lw-Count, an encoder-decoder model developed by Liu et al., employs efficient, lightweight convolutional modules and a scale regression module to minimize artifacts and achieve high accuracy at low computational complexity [37]. In congested scenes, Liang et al. introduced PDDNet, which uses GhostNet layers and depthwise dilated convolutions to extract multi-scale features [38] efficiently. Khan et al. proposed LCDnet, a very small CNN (0.05M parameters) with small, separable convolutions and curriculum learning to count objects in real time on a drone. Lee et al. proposed TinyCount, a MobileNetV2-based model with approximately 60K parameters, SE blocks, and a scale-perception module, for fast and accurate inference on edge devices [39].

These lightweight crowd-counting models suggest that effective feature extraction, depth processing, a small back-end, and attention/dynamic processing can greatly reduce computation with little regard for performance. Such developments enable in-the-field, real-time mobile operations, and architectures of a lightweight nature are required for feasible, scalable crowd-monitoring systems. However, the fact remains that, whatever lightweight crowd-counting models are, they are heavily based on simplifying architectural and attention processes, and people do not emphasize internal knowledge transfer tactics that can reduce low model capacity. This is why compact architectures still struggle to be robust and spatially coherent in very dense scenes.

2.3. Knowledge Distillation

One of the most commonly used model compression methods is knowledge distillation (KD). It aims to enhance the performance of deep neural networks by transferring knowledge of a high-capacity teacher model to a small student model [15]. KD can use small models to maintain semantic expression and generalization at minimal computational cost via soft predictions and intermediate feature supervision. KD is especially useful in the context of crowd counting because of its high-density, regression-based density map estimation, which requires maintaining contextual awareness and spatial continuity. Early KD-based crowd-counting solutions were based on structured feature transfer, such as Structured Knowledge Transfer (SKT), which compresses intra- and inter-layer relationship information into feature patterns to guide lightweight networks toward useful spatial representations [17]. Nevertheless, traditional teacher-student distillation systems are frequently affected by capacity and error-propagation problems, in which teacher-inaccurate predictions negatively impact student learning [40].

To address these issues, review-based and task-specific distillation procedures were proposed to refine transferred knowledge and prevent detrimental supervision during training [41]. Recent literature has suggested dual-stage and weighted distillation models that combine global, local, and feature-level directions, which are found to be more robust and generalize to dense, highly populated crowds at the expense of computational speed [42], [43]. Furthermore, KD can be extended to multimodal problems in crowd counting, e.g., RGB-thermal and drone-based tasks, where hierarchical, hybrid, and collaborative distillation can be employed to create lightweight student models that achieve parameter and inference latency reductions [44], [45]. Collectively, they are publications that solidify the idea of knowledge distillation as a facilitating technique for achieving a precise and efficient approach to crowd counting in real time and in resource-constrained environments, and that introduce the novel complexities and dependencies of external teacher-based systems [46]. These teacher-student distillation models have some overhead, are more effective, but are more sensitive to extra teacher training and architecture-dependent, and can introduce teacher-specific spatial bias or density-map artifacts, which are undesirable when learning fine-grained density regression in crowded scenes.

2.4. Self-Knowledge Distillation

Self-Knowledge Distillation (SKD) is a teacherless learning model that enhances extrapolation by self-monitoring through self-predictions or internal representations. Initial experiments demonstrated that stabilizing and reducing overfitting caused by normalizing predictions by class or by reusing the softened self-outputs can be achieved without increasing the model's complexity [47], [48]. Later on, SKD was generalized to feature-level refinement, in which lower levels receive more and more semantic information from higher levels, acquiring it through learning more discriminative features [49], [50]. Alternative strategies for progressive target refinement were proposed to prevent confirmation bias and ensure that the model can update its self-generated supervision during training [51]. and more recently, single-unified formulations showed that SKD can be considered a special case of classical knowledge distillation, achieving similarly good performance improvements without requiring an external teacher network [52].

All these studies make it clear that LSKD is an effective and scalable solution to enhancing lightweight models with limited training and deployment requirements. Even though the majority of self-knowledge distillation strategies were initially developed for classification or recognition, their focus on internal consistency per layer makes them especially well-suited to density-based crowd counting, where the preservation of spatial coherence and higher-level feature alignment are paramount. Unlike current methods, this paper combines lightweight architecture design with self-knowledge distillation, specifically for density map regression. Unlike traditional teacher-student distillation systems, the proposed approach enables knowledge transfer between its internal nodes via a single, small network thereby enhancing feature consistency and stability without requiring additional inference steps. This makes it a realistic and scalable solution for real-time crowd counting on constrained-resource platforms.

3. Proposed Method

3.1. Overview of the Network

The proposed solution is a regression-based crowd counting approach that produces a continuous density map from the input image, and the resulting crowd count is calculated by summing the density map values across all pixels. The main goal of network design is to be as precise as possible while remaining lightweight and computationally efficient so that it can be used in a real-world implementation. Similar to the large plan shown in Figure 1, the input image is first processed by a light convolutional backbone, which also provides representations at different

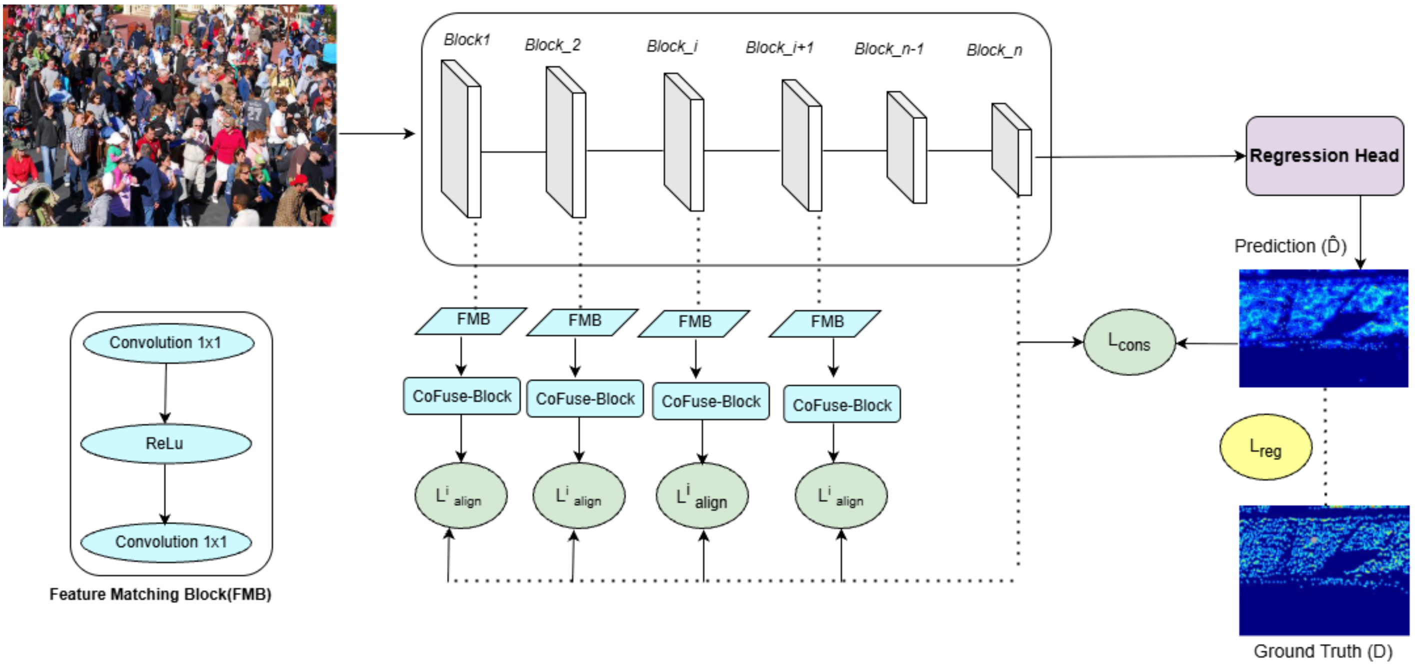

Figure 1. Overall architecture of the proposed LSKD framework, showing the lightweight backbone, feature matching and context fusion modules, and the final density map prediction.

scales. These multi-level features encode complementary data: the earlier levels represent fine-scale spatial data (edges and local textures), and the later layers encode high-level semantic data and larger-scale patterns in the context of crowded, highly cluttered scenes. The richest feature representation is then fed into a regression head to produce the final density map prediction, the model's primary outcome. To further strengthen the learning process without increasing the network's inference-time complexity, the network under analysis uses a self-knowledge distillation strategy during training. This model uses its own more enriched representations as internal guidance cues for intermediate stages, forcing shallower representations to become more semantically informative without losing their spatial sensitivity.

To achieve this internal distillation, intermediate features are first passed through Feature Matching Blocks (FMB), which align feature representations, and then supervision is applied. Moreover, the framework proposes the Context Fusion (CoFuse) block, a global context fusion module that injects contextual details into intermediate representations. This enhances the uniformity of middle features and helps the network handle significant changes in appearance, perspective distortion, and dense crowds. In the general architecture, branches of intermediate supervision tasks work with the primary regression path and provide signals for multi-level learning, but do not lighten the final inference path.

The guided training used by this training is based on the composite goal that sums up the loss components as expressed in the architecture: intermediate features-level loss, the loss that is used in supervision of self-distillation, a consistency loss, the loss that is used in supervision of the relationships between the deep representations and density prediction, and a direct loss, which is the loss that is used in supervision of the final density map output. These elements come together to create a robust end-to-end architecture in which context modelling at the global scale and feature correspondence stimulate internal knowledge action, ultimately leading to improvements in the quality of density estimates without the need for a dense external teacher model.

3.2. Feature Matching Block (FMB)

Figure 1 represents the FMB which is a light adapter that adapts the intermediate backbone capability to the attribute of deep projector used in the mechanism of self-knowledge distillation. Intermediate and deep features cannot be directly matched since they have different channel dimensions and are sensitive to different semantic levels and this can compromise distillation signal. FMB takes this care by projecting all the intermediate feature maps using two 1x 1 convolutions and a ReLU in between. The first 1 × 1 convolution converts the intermediate feature to the same channel as the deep feature, the second 1 × 1 convolution brings non-linearity, and the last 1 × 1convolution converts the transformed feature to create the aligned feature on which supervision is used. Mathematically the aligned feature of an intermediate feature H_i is found as:

$$\widetilde{H}_i = Conv_{1\times1}\left(ReLU\left(Conv_{1\times1}(H_i)\right)\right) \tag{1}$$

and the constraint of the distillation is provided by the minimization of the difference between the deep projector feature H_g and the aligned feature $\widetilde{H}_i$.

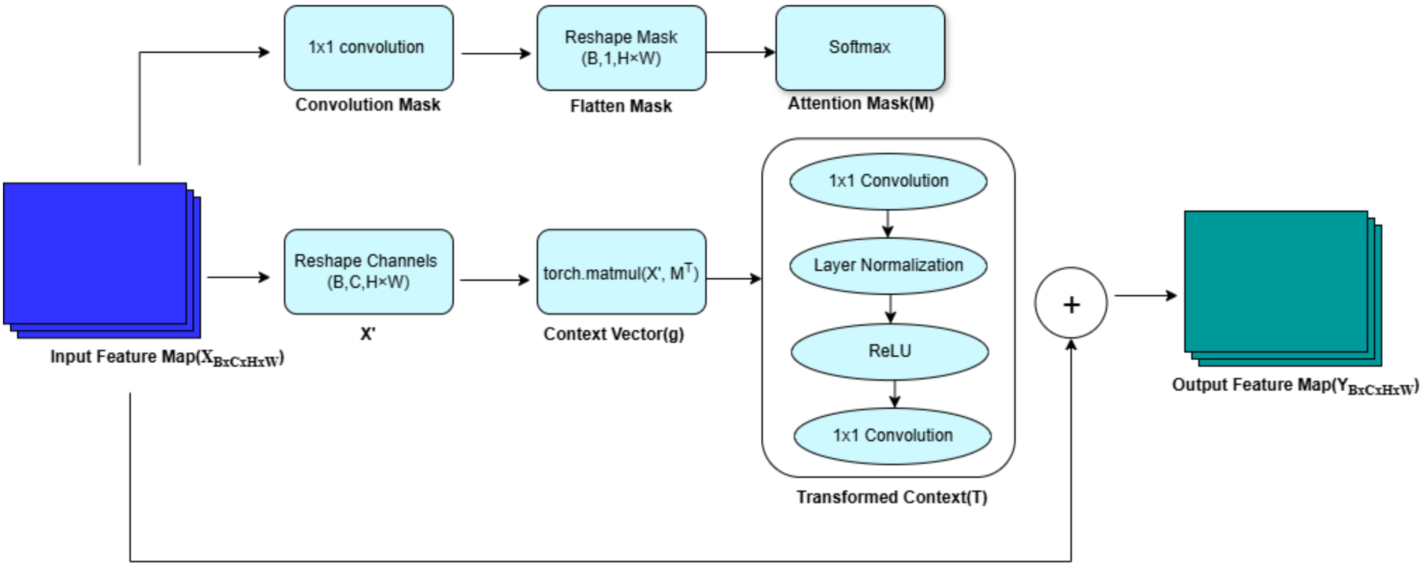

Figure 2. CoFuse block generates a spatial attention mask to pool a channel-wise global context descriptor, refines it through a lightweight transform, and fuses it back into the feature map by residual addition.

3.3. Context Fusion Block (CoFuse)

As shown in Figure 2, CoFuse is a lightweight global-context module that enhances an intermediate feature map by injecting scene-level context at every spatial map location. This may be especially handy when the count of a dense crowd is needed, and the local convolutional evidence is often unclear due to high occlusion and large dimension variance. CoFuse takes an input feature map, $X \in R^{B\times C\times H\times W}$and outputs a feature map $Y \in R^{B\times C\times H\times W}$ which are of the same size, where B is the batch size, C is the number of channels and $H \times W$ is the spatial resolution. CoFuse first computes a spatial score map which is a 1×1 convolution thus giving a single score at each spatial position:

$$S = \mathrm{Conv}_{1\times1}(X), \quad S \in R^{B\times\mathbb{1}\times H\times W} \tag{2}$$

The score map S is then flattened with the spatial dimensions to length HW and SoftMax is then done at each of the HW positions to give an attention mask $M \in R^{B\times\mathbb{1}\times(HW)}$ where each element is the weighted significance of a spatial position:

$$M_{b,1,p} = \frac{\exp\left(S^f_{b,1,p}\right)}{\sum_{q=1}^{HW} \exp\left(S^f_{b,1,q}\right)}, \quad p = 1, \dots, HW \tag{3}$$

Where S^f is the flattened form of S and p is an index that corresponds to a different location in space after the flattening has taken place.

Simultaneously, the input feature map X is flattened to $X' \in R^{B\times C\times(HW)}$. CoFuse uses the attention mask to compute a small global context vector $g \in R^{B\times C\times 1}$, that is the weighted average of the spatial positions (equivalent to the matrix multiplication operation in the diagram):

$$g = X'M^{\top}, \quad g \in R^{B\times C\times\mathbb{1}} \tag{4}$$

In this case $X'$is the flattened input feature map and $M^{\top} \in R^{B\times(HW)\times\mathbb{1}}$ is the attention mask transpose. The context value generated in each channel of this operation summarizes the whole feature map based on the learned spatial attention.

The transformed context unit then refines the context vector as shown in the figure, which is a lightweight bottleneck made up of 1×1 convolution, Layer Normalization, ReLU, and a second 1×1 convolution:

$$T = \mathrm{Conv}^{(2)}_{1\times1}\left(\mathrm{ReLU}\left(\mathrm{LN}\left(\mathrm{Conv}^{(1)}_{1\times1}(g)\right)\right)\right), \quad T \in R^{B\times C} \tag{5}$$

The two 1x1 convolution layers are referred to as the $\mathrm{Conv}^{(1)}_{1\times1}$ and $\mathrm{Conv}^{(2)}_{1\times1}$ and these two layers are used to transform the context vector. $LN(\cdot)$ represents the Layer normalization and $ReLU(\cdot)$ is the activation function. T is the transformed context vector which gives one refined context value in each channel.

Lastly, the residual map is reconstructed to the original feature map by a fusion of the transformed context. As there is only one value per channel, the context value in that channel is channel-wise repeated across all spatial locations in the channel:

$$Y_{b,c,h,w} = X_{b,c,h,w} + T_{b,c} \tag{6}$$

In this case $T_{b,c}$ is added to the feature map on a channel-by-channel basis. All spatial locations (h, w) in the channel have the same value of context that will be added to the channel. This creates the output feature map Y which is the same size as X but with residual fusion of contextual information across the entire picture.

In conclusion, CoFuse learns a spatial attention mask to find an informative location, aggregates global context, downsizes it with a simple nonlinear transform, and applies it to the feature representation. Unlike more traditional channel-only attention mechanisms, such as SE blocks or attention modules, CoFuse combines spatial relevance with channel-wise global context via a simple attention pooling mechanism. CoFuse can optimistically boost intermediate representations, yielding global context descriptors, without incurring the high computational cost of an attention head or the complexity of inference by generating a spatial attention mask to compute a global context descriptor and then reinjecting it via residual fusion.

3.4. Self-Knowledge Distillation (SKD)

To improve the representation learning process that does not rely on an external teacher network, we apply a self-knowledge distillation method that uses deep network features to teach low-level layers during training. It allows the richer representation of semantic information, as well as allows the light weight of the model in the inference stage by using intermediate representations as a way of regularizing against drift in representations as we go deeper in the network, where optimization stability and encouraging intermediate layers to learn semantically but spatially sensitive features are necessary in problems of dense regression.

According to an input image I, the backbone network obtains a collection of in-between feature maps $\{\mathrm{H_i}\}_{\mathrm{i=1}}^{\mathrm{N}}$, and a deep feature map H_g of the last stage. The number of intermediate layers involved in distillation of self-knowledge is denoted by N. The representation of each middle feature map is first subject to a Feature Matching Block (FMB) in order to match its representation with the deep feature:

$$\widetilde{H}_i = \mathrm{FMB}(H_i) \tag{7}$$

In this case, H_i is the intermediate feature map of the i-th backbone stage, Feature Matching Block is denoted as FMB, and the aligned intermediate representation following channel and feature adjustment is denoted as $\widetilde{H}_i$. The aligned feature is further optimized with CoFuse block to bring global contextual information:

$$H_i^c = \mathrm{CoFuse}\left(\widetilde{H}_i\right) \tag{8}$$

The operator CoFuse(·) injects global context into the aligned feature, producing the context-enhanced intermediate feature H_i^c.

In order to acquire training of deep semantic knowledge, feature alignment loss is used:

$$L_{align}^i = ||H_i^c - H_g||^2 \tag{9}$$

$||\cdot||^2$ is the mean squared error. L_{align}^i computes the difference between deep feature map H_g and the context enhanced intermediate feature H_i^c using the mean squared error.

The total amount of all the intermediate losses is then brought to one loss, intermediate alignment loss:

$$L_{IA} = \frac{1}{N}\sum_{i=1}^{N} L_{align}^i \tag{10}$$

where L_{IA} denotes the overall intermediate alignment loss. A consistency constraint L_{cons}, between the deep feature and the density map on which the forecast is being projected, is added between the two to project the deep representations to the final forecast:

$$L_{cons} = ||\mathrm{Avg}\left(H_g\right) - \widehat{D}||^2 \tag{11}$$

The operator Avg(·) performs channel-wise averaging on the deep feature map H_g,yielding a density-like representation that is compared with the predicted density map $\widehat{D}$. Direct supervision on the predicted density map is provided using the ground-truth density map D through a combination of pixel-wise error and structural similarity as L_{reg}:

$$L_{reg} = \lambda_1||\widehat{D} - D||^2 + \lambda_2\left(1 - \mathrm{SSIM}\left(\widehat{D}, D\right)\right) \tag{12}$$

The weighting factors λ_1 and λ_2 balance numerical accuracy and structural consistency in the density estimation.

The final training objective combines all loss terms as L_{total}:

$$\mathrm{L}_{total} = \alpha\, L_{IA} + \beta\, L_{cons} + \gamma\, L_{reg} \tag{13}$$

with α, β and γ controlling the contribution of each component during optimization.

Overall, this self-knowledge distillation strategy enables effective information transfer from deep to intermediate layers during training, leading to improved density estimation performance without introducing additional parameters or computational overhead at inference time.

4. Experimental Setup

4.1. Datasets

Three popular public crowd counting benchmarks (ShanghaiTech, UCF-QNRF, UCF-CC-50) are evaluated using the proposed method. As ShanghaiTech consists of Part A and Part B, there will be four evaluation subsets in the overall analysis. The datasets can be easily used to assess accuracy and robustness, as they are diverse in terms of crowd type, crowd density, perspective distortion, and annotation difficulty, and are frequently used as reference benchmarks for both accuracy and efficiency in crowd counting.

The main statistics for each dataset are summarized in Table 1 which includes the number of images and the

Table 1. Benchmark dataset statistics, where Dens is density level, Res is image resolution, N is number of images, C is total annotations, and Min, Max, Avg denote crowd count distribution.

Dataset	Dens	Res	N	C	Min	Max	Avg
ShanghaiTech A	Low-D	868 × 589	482	~240,000	33	3139	501
ShanghaiTech B	Low-D	1024 × 768	716	~90,000	9	578	123
UCF-QNRF	Mid-D	2902 × 2013	1535	~1.25 million	49	12865	815
UCF-CC-50	Ultra-High-D	2888 × 2101	50	~64,000	94	4543	1279

Table 2. Crowd density-level distribution of the evaluated datasets, showing the count ranges for low, medium, high, and very high crowd scenes.

Dataset	Low	Medium	High	Very High
ShanghaiTech A	1 - 9	10 - 21	22 - 47	>= 48
ShanghaiTech B	1 - 2	3 - 4	5 - 10	>= 11
UCF-QNRF	1 - 10	11 - 30	31 - 78	>= 79
UCF-CC-50	1 - 20	21 - 57	58 - 121	>= 122

diversity of crowds across datasets in terms of both size and challenge. Moreover, Table 2 provides the population-level distribution across four datasets, which show different levels of crowd congestion: sparse, moderate, and very dense. The density- level thresholds are defined following prior crowd counting studies, where images are grouped according to the total crowd count to reflect increasing levels of congestion and occlusion. A combination of these two tables provides a clear picture of the dataset's features and highlights the challenges posed by the various distributions of crowd density.

4.1.1. ShanghaiTech Part A

ShanghaiTech Part A contains 482 crowd images with nearly 240,000 annotated persons. The pictures are taken randomly, from the Internet, and show numerous crowded situations in which people tend to crowd each other and may appear at different sizes due to perspective. This complicates counting due to heavy occlusion and large variations in crowd density. The data will be split into 300 training images and 182 test images.

4.1.2. ShanghaiTech Part B

ShanghaiTech Part B has nearly 90,000 annotated street images, with crowds relatively thin compared to Part A. It has 400 training images and 316 test images. The scenes are less varied and more composed, yet counting remains difficult since there are people who are shown at varying sizes because of alterations in the camera angle and the distance.

4.1.3. UCF-QNRF

UCF-QNRF represents one of the most difficult and the most significant datasets of the number of people in the crowd with nearly 1.25 million annotated individuals. It is constituted of the 1,535 quality crowd images gathered variously on the web. The data set has numerous contrasting scenes in terms of camera movement, lighting and density of the crowd. The size of the number of people varies between 49 to 12,865 and this makes it more realistic and challenging. Since the pictures are of a high resolution, the sizes of the heads differ significantly, which contributes to even more complexity in terms of proper counting.

4.1.4. UCF-CC-50

UCF-CC-50 is very challenging, but a small crowd counting dataset containing just 50 pictures and a total of nearly 64,000 annotated persons. The crowd in the scenes is tremendous, and there are between 94 and 4,543 people in every picture, with a mean figure of approximately 1,280. The photos contain great changes of perspective that cause the size of the heads to change significantly and make counting harder. It is commonly tested on 5-fold cross-validation due to the very small size of the dataset, and most deep models cannot generalize effectively due to the low amount of training data.

4.2. Implementation Details

We follow a standard density-regression training pipeline with dataset-specific cropping and carefully balanced supervision and distillation losses. During training, images are randomly cropped to a fixed size and horizontally flipped for data augmentation. Since ShanghaiTech Part A contains comparatively lower-resolution images, we use 256 × 256 crops for this dataset. For ShanghaiTech Part B, UCF-CC-50, and UCF-QNRF, we adopt larger 512 × 512 crops to preserve more contextual information.

Ground-truth (GT) density maps are constructed by convolving each head annotation with a fixed Gaussian kernel of size 15. Although adaptive Gaussian kernels can better model perspective variation, we adopt a fixed kernel size to maintain consistency across datasets and ensure fair comparison with lightweight baselines, while avoiding additional pre-processing overhead.

All the experiments are done with the help of the PyTorch framework and are run on an NVIDIA GeForce RTX 4080. We apply the Adam optimizer to optimize the network and train all models in 150 epochs with a batch size of 2. The learning rate starts with the value of 1 × 10⁻⁴ and is reduced in the later stages of training to enhance convergence stability with a lower learning rate in the later stages of fine-tuning in the final epochs.

To make sure that training is mainly pushed by the ground-truth density regression and also takes the advantage of self-distillation as a regularizer, we set the loss weights to be as follows α = 6.0, β = 2.0, and γ = 13.0 in the total loss. To avoid dataset-specific tuning and provide fair evaluation, the loss weights were not tuned on a case-by-case basis and were fixed throughout all datasets.

This configuration maintains hard supervision as the dominant signal, allows intermediate distillation to provide meaningful semantic guidance without over-constraining spatial details, and employs projector consistency as a lightweight stabilizer. For the hard loss, we set λ_1 = 1.0 and λ_2= 0.2, emphasizing numerical accuracy through the mean squared error (MSE) term while incorporating a modest structural similarity (SSIM) component to enhance the structural coherence of the predicted density maps without degrading regression performance.

4.3. Evaluation Metrics

4.3.1. Counting Accuracy Metrics

To evaluate counting performance, we adopt two widely used metrics in crowd counting: **Mean Absolute Error (MAE)** and **Mean Squared Error (MSE)**. Following standard crowd counting benchmarks, both metrics are computed at the **image-level count difference**, rather than at the pixel level of density maps. In density-based crowd counting, the predicted total count for each image is obtained by integrating (summing) the predicted density map over all spatial locations. Similarly, the ground-truth count is computed by summing the ground-truth density map. Therefore, for a test set containing N images, let $D_i^{\text{pred}}(x,y)$ and $D_i^{\text{gt}}(x,y)$ denote the predicted and ground-truth density values at spatial location (x,y) for the i-th image. The corresponding predicted and ground-truth crowd counts are defined as:

$$C_i^{\text{pred}} = \sum_{x,y} D_i^{\text{pred}}(x,y) \tag{14}$$

$$C_i^{\text{gt}} = \sum_{x,y} D_i^{\text{gt}}(x,y) \tag{15}$$

The evaluation metrics are then computed as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|C_i^{\text{pred}} - C_i^{\text{gt}}\right| \tag{16}$$

$$MSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(C_i^{\text{pred}} - C_i^{\text{gt}}\right)^2} \tag{17}$$

MAE measures the average absolute deviation between predicted and ground-truth counts, reflecting overall counting accuracy. MSE penalizes larger deviations more heavily and thus reflects the robustness and stability of the model across varying crowd densities.

4.3.2 Computational Complexity Metrics

To compare the effectiveness of the proposed model with two popular indicators, the number of parameters (Params) and the complexity of computations (FLOPs) and overall results are given in Table 4. For a 2D convolution layer, the number of learnable parameters depends on the kernel size, the input/output channel dimensions, and the bias term. It can be written as:

$$P_{\text{conv}} = (r\,t\,C_{\text{in}} + 1)C_{\text{out}} \tag{18}$$

where r and t denote the kernel height and width, C_{in} and C_{out} are the input and output channel dimensions, and the constant +1 accounts for the bias.

The computational cost of the same layer is measured in FLOPs and is expressed as:

$$F_{\text{conv}} = 2\,H_{\text{out}}\,W_{\text{out}}(r\,t\,C_{\text{in}} + 1)C_{\text{out}} \tag{19}$$

Where H_{out} and W_{out} are the spatial height and width of the feature map output. The factor of 2 is in keeping with the traditional system that a multiply-accumulate operation comprises one multiplication and one addition.

5. Results and Analysis

5.1. Quantitative Results

It is important to mention that the compared techniques utilize various architectures, supervision techniques, and backbone complexity, such as heavy CNNs, transformer-based models, and detection-assisted approaches. It is a comparison that is made to show general trends in performance across performance and efficiency trade-offs, and not to assert any direct superiority in performance to models with significantly larger computational budgets.

Table 3 [37], [53], [54] shows the quantitative results of comparing the proposed LSKD to a large variety of the state-of-the-art crowd counting methods in ShanghaiTech Part A, ShanghaiTech Part B, UCF-QNRF, and UCF-CC-50. The highest results in each column are highlighted in bold. The table also gives the number of parameters in the model to be analyzed in order to analyze the trade-off between counting accuracy and the complexity of the model.

LSKD performs competitively across all four datasets while maintaining a lightweight structure. LSKD has 63.17

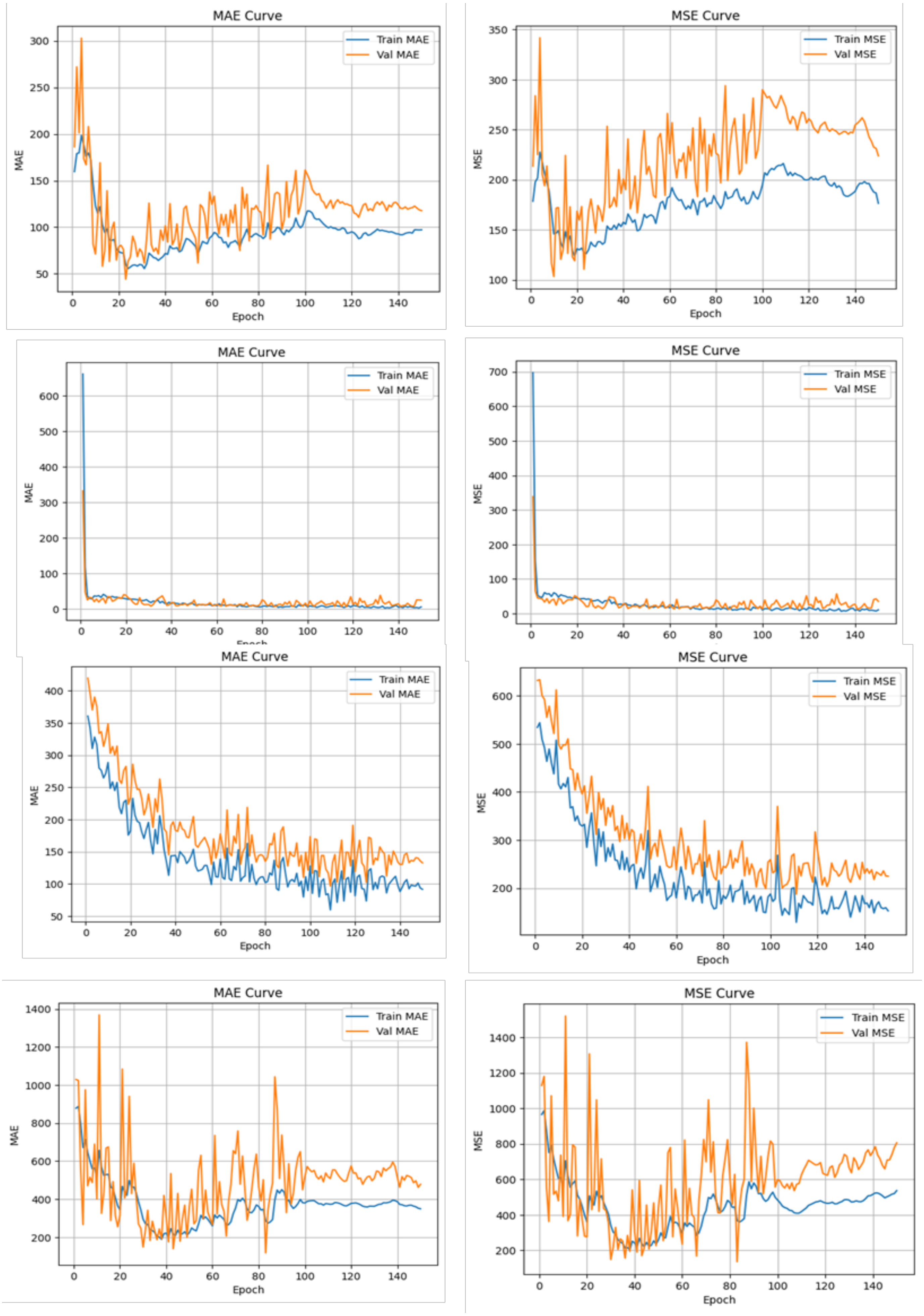

Figure 3. MAE and MSE curves for training and validation datasets.

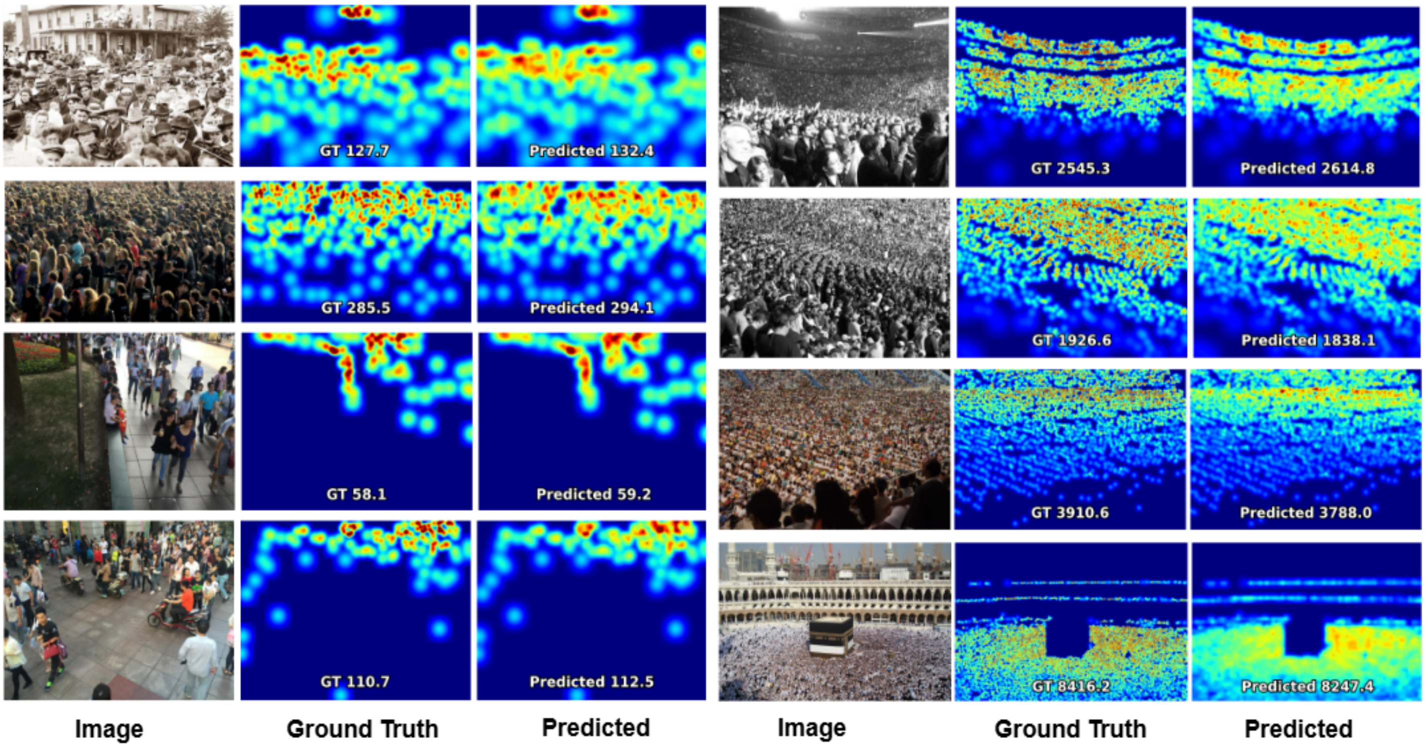

Figure 4 Visualization results of the proposed LSKD method. The left panel shows examples from ShanghaiTech Part A and B, while the right panel presents results from UCF-QNRF and UCF-CC-50. For each dataset, the input image (left), ground-truth density map (middle) and predicted density map(right) are displayed.

MAE and 102.95 MSE on the ShanghaiTech Part A dataset, respectively, compared with a few existing methods that use much more parameters (2.65M) than heavy models like LSC-CNN (42.23M parameters) and CCST (294.7M parameters).

LSKD achieves 8.94 MAE and 11.51 MSE on the ShanghaiTech Part B dataset, indicating strong accuracy for a small model with a rather sparse crowd. LSKD scores 143.7 MAE and 239.2 MSE on the difficult UCF-QNRF dataset, which encompasses profound changes in crowd density, image quality, and scene complexity. Although some new approaches achieve lower error rates, they often rely on much larger models, which are more expensive to compute. In contrast, LSKD provides a competitive balance between accuracy and cost. On the contrary, LSKD maintains constant performance at the expense of being lightweight, making it robust and having high generalization capacity. For the very dense UCF-CC-50 data set, with very few training samples, the LSKD results are 223.88 MAE and 341.72 MSE. Although this dataset is not simple, the suggested approach is competitive and effectively works with ultra-high-density scenes without relying on parameter-intensive structures. Figure 3 demonstrates the development of MAE and MSE of the training and validation sets of the four benchmark datasets. The first row is ShanghaiTech Part A, the second is ShanghaiTech Part B, the third is UCF-QNRF, and the fourth is UCF-CC-50, indicating the model's error dynamics across each dataset.

On the whole, these findings show that LSKD allows offering a good balance between accuracy and efficiency, in which consistent results are recorded in both sparse and extremely dense situations, with the model size being small enough to be used in practice. Over repeated runs and across cross-validation folds, the convergence behavior of the proposed method is stable with minimal variation in MAE and MSE, which indicates strong performance in both sparse and extremely dense scenes. Visual comparison of the density maps, as illustrated in Figure 4, suggests that the proposed method is capable of yielding consistent results at scale, despite the limited training data in the ultra-dense scenarios. The projected maps show reduced error activation in the background areas and more accurately maintain spatial continuity in the highly congested areas, which indicates the power of global context fusion and self-distillation.

It is mandatory to mention that some of the lightweight approaches in Table 3 show competitive results using much fewer parameters. As an example, LSANet (0.2M parameters) has 8.6 MAE on ShanghaiTech Part B, and Lw-Count (0.071M parameters) also shows good efficiency-accuracy trade-offs. These outcomes imply that strong performance can be achieved through careful lightweight architectural design. Nevertheless, most methods primarily focus on architectural simplification, but the suggested LSKD framework is also concerned with internal feature alignment and learning contextual consistency without supervision. The comparison thus brings in the various design philosophies and not a performance hierarchy.

Table 3. Comparison of the proposed LSKD method with state-of-the-art crowd counting approaches on ShanghaiTech Part A and B, UCF-QNRF, and UCF-CC-50 in terms of MAE and MSE.

Method	Venue	Parameters (M)	Shanghai Part A		Shanghai Part B		UCF-QNRF		UCF-CC-50	
			MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN	CVPR16	0.13	110.2	173.2	26.4	41.4	277.0	426.0	377.6	509.1
CMTL	AVSS17	2.46	101.3	152.4	20.0	31.1	252.0	514.0	322.8	397.9
CP-CNN	ICCV17	68.4	73.6	106.4	20.1	30.1	-	-	295.8	320.9
SANet	ECCV18	1.39	67.0	104.5	8.4	13.6	152.6	247.0	258.4	334.9
CSRNet	CVPR18	16.26	68.2	115.0	10.6	16.0	145.5	233.3	266.1	397.5
ACSCP	CVPR18	5.10	75.7	102.7	17.2	27.4	-	-	291.0	404.6
SaCNN	WACV18	24.06	86.8	139.2	16.2	25.8	-	-	314.9	424.8
LCN	ICIP19	0.032	93.3	157.0	15.1	23.3	262.0	358.6	262.0	358.6
PCC Net	TCSVT19	0.55	73.5	124.0	11.0	19.0	148.7	247.3	240.0	315.5
BL	ICCV19	21.50	61.5	103.2	7.5	12.6	87.7	158.1	229.3	308.2
MobileCount	NEUCOM20	3.40	89.4	146.0	9.0	15.4	131.1	222.6	321.7	437.1
ASNet	CVPR20	30.39	57.8	90.0	-	-	91.6	159.7	174.8	251.6
LSC-CNN	TPAMI20	42.23	66.5	101.8	7.7	12.7	120.5	218.2	-	-
DM-Count	NIPS20	-	59.7	95.7	7.4	11.8	85.6	148.3	211.0	291.5
UEPNet	ICCV21	26.21	54.6	91.2	6.4	10.9	**81.1**	**131.7**	165.2	275.9
P2PNet	ICCV21	18.34	**52.7**	**85.1**	6.3	9.9	85.3	154.5	172.7	256.1
DKPNet	ICCV21	30.63	55.6	91.0	6.6	10.9	81.4	147.2	-	-
SCNet	T-AES21	16.3	58.5	99.1	8.5	13.4	93.9	150.8	197.0	231.6
CFANet	WACV21	-	56.1	89.6	6.5	10.2	89.0	152.3	203.6	287.3
AECNet	CAFGR21	-	55.2	92.4	**5.12**	**9.6**	84.2	142.7	**152.8**	251.6
Lw-Count	TCSVT22	0.071	69.7	100.5	10.1	12.4	149.7	238.4	239.3	307.6
LSANet	NEUCOM22	0.2	66.1	110.2	8.6	13.9	112.3	186.9	-	-
STNet	TMM22	15.56	52.9	83.6	6.3	10.3	87.9	166.4	162.0	**230.4**
JANet	NEUCOM22	21.77	53.9	90.5	6.3	10.4	90.0	168.6	-	-
CCST	VCOMP24	294.7	62.8	94.1	8.3	13.4	93.7	166.9	190.7	289.0
LSKD (ours)	-	**2.65**	63.17	102.95	8.94	11.51	143.7	239.2	223.88	341.72

Table 4. Computational complexity comparison of the proposed LSKD model with existing crowd counting methods in terms of parameter count (Params) and floating-point operations (FLOPs).

Methods	Parameters(M)	FLOPs(G)
MCNN	**0.13**	21.17
CMTL	2.46	95.56
SANet	1.39	71.46
PCC Net	0.55	129.58
MobileCount	3.40	6.15
LSANet	0.2	**5.41**
CSRNet	16.26	325.02
CANet	18.10	344.47
SFCN	38.6	486.1
LSKD (ours)	2.65	10.23

5.2. Efficiency Analysis

Table 4 presents the computational complexity comparison between the proposed LSKD model and representative crowd counting approaches. The proposed method requires only **2.65M parameters** and **10.23 GFLOPs**, demonstrating its lightweight design. Compared with heavy models such as CSRNet (16.26M parameters, 325.02 GFLOPs) and SFCN (38.6M parameters, 486.1 GFLOPs), LSKD significantly reduces computational cost while maintaining competitive counting accuracy. Compared to lightweight alternatives such as LSANet (0.2M parameters, 5.41 GFLOPs), LSKD provides improved accuracy with a moderate increase in complexity, indicating a favorable efficiency–accuracy trade-off. These results confirm that the proposed self-knowledge distillation framework enhances feature representation without introducing substantial inference overhead, making it suitable for real-time and resource-constrained deployment scenarios.

5.3. Ablation Study

Ablation experiments are used to examine the contribution of each of the components in the proposed method. We create a baseline model and add different modules one by one and test their impact, i.e., counting performance. All the experiments are measured using MAE and MSE

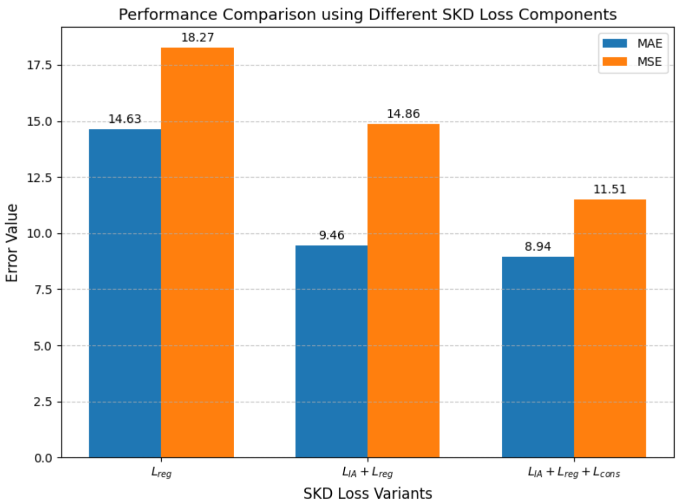

Figure 5. Ablation on effect of different SKD loss components.

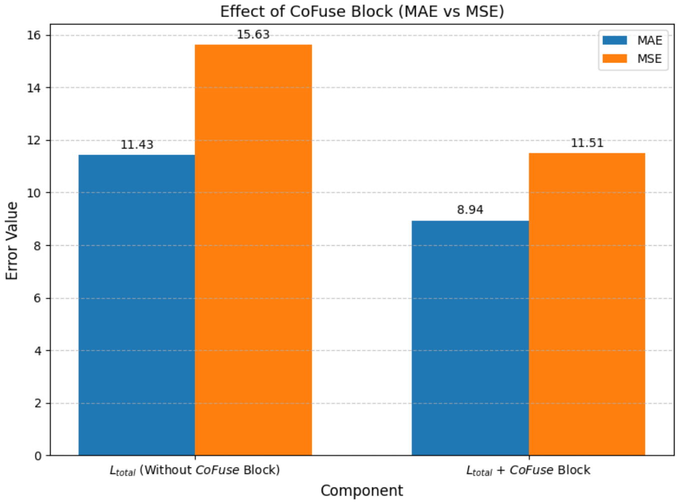

Figure 6. Ablation on the effect of CoFuse Block.

using the ShanghaiTech part B dataset. ShanghaiTech Part B will be chosen as an ablation because it provides a moderate evaluation condition both in terms of persons and spacing, and certain training data, which allows for adjusting assessments of the architectural and loss factors, which can be traced in the stable gains of the overall benchmark outcomes.

The given research helps to understand how each design choice affects the final performance. The lightweight backbone and the final density regression head are the elements of the base model that are being trained on the hard supervision loss without self-knowledge distillation modules or context fusion modules.

5.3.1. Effect of SKD Losses

In order to test the role of various elements of SKD, we perform ablation experiments on the ShanghaiTech Part B. The findings are represented using Figure 5.

When the model is trained using only the hard supervision loss $\mathrm{L}_{\mathrm{reg}}$ it achieves 14.63 MAE and 18.27 MSE, indicating limited performance when relying solely on direct density regression. By introducing the intermediate distillation loss L_{IA} alongside $\mathrm{L}_{\mathrm{reg}}$ the performance improves significantly to 9.46 MAE and 14.86 MSE, demonstrating that intermediate supervision provides effective guidance for learning more discriminative representations. Compared to the baseline, introducing intermediate distillation reduces MAE by 35.3% and MSE by 18.7%.

Finally, when all loss terms are enabled, including the projector consistency loss L_{cons} , the model achieves the best performance with 8.94 MAE and 11.51 MSE corresponding to a 38.9% reduction in MAE and a 37.0% reduction in MSE compared to the baseline. This confirms that each SKD loss component contributes positively, and their combination leads to consistent and complementary performance gains. The projector consistency loss encourages alignment between deep semantic activations and the final density prediction, reinforcing structural consistency and reducing semantic–spatial mismatch during training.

5.3.2. Effect of CoFuse Block

We also conduct ablation experiments to test the efficacy of the proposed CoFuse block, which can also be visualized in Figure 6. The CoFuse block in the model is taken out, resulting in a significant decrease in performance to 11.43 MAE and 15.63 MSE, which means that the network is not getting enough global contextual information.

Once the CoFuse block is introduced into the model, it has a distinct performance increase with the error decreasing to 8.94 MAE and 11.51 MSE. This represents a decrease of 21.8% and 26.4% in MAE and MSE, respectively. These findings show that CoFuse is effective in improving intermediate feature representations by injecting global contextual features to achieve more accurate and robust density estimation. In addition, the enhancement implies that CoFuse assists the network in overcoming the issue of ambiguous local evidence in the presence of occlusion and large-scale variation in complex crowd scenes.

All other components, such as the backbone, SKD losses, and training settings, are held constant in this ablation, so that the effects of the CoFuse block are solely due to its presence. It is noteworthy that the beneficial effects of the CoFuse block are greater in combination with self-knowledge distillation, which indicates that global context enhancement and internal feature alignment are synergistic factors in enhancing density estimation. Although the ablation study controlled a large number of components, including the backbone, these results are also consistent with the entire benchmarking results reported in Section 4.

6. Conclusion

LSKD is a simple and efficient crowd-counting model showing that internal self-knowledge distillation may assist in addressing model capacity limitations in the framework of density map regression, without having to resort to the usage of an external teacher network. Under self-knowledge distillation, the model subsequently applies the deep representations into the intermediary representations in training by aligning multi-level features with an FMB block, infused global contextual awareness with the CoFuse module, and the proposed framework can also implement semantic guidance to the intermediary representations successfully and maintain spatial sensitivity under a high level of occlusion conditions, scale variations, and crowd densities. It has been proved that the LSKD provides an appropriate quality-cost ratio, which is achieved by the balance, that is why the given approach is a good option that must be applied in resource-limited settings in real time, when the size of the model is one of the most important aspects to be considered, as well as the computational budget.

Future studies could further enhance the framework by exploring more hardware-efficient lightweight backbones, developing more robust and stable self-distillation objectives, and improving deployment on real edge devices through techniques such as pruning and quantization. In addition, the method could be extended to video-based crowd counting by incorporating temporal consistency into its predictions, enabling more stable performance in real-world surveillance systems. Although LSKD is an effective and efficient approach, there is still room for improvement. Future work may focus on improving performance in extremely sparse scenes and in scenarios with severe perspective distortion, where lightweight density regression models may still face challenges due to limited model capacity.

7. Declarations

7.1. Author Contributions

Muhammad Raza: Conceptualization, Methodology, Software, Data Curation, Formal Analysis, Investigation, Visualization, Writing – Original Draft; **Miaogen Ling:** Supervision, Conceptual Guidance, Research Design Oversight, Critical Review & Editing, Project Administration; **Atta Ur Rahman:** Formal Analysis, Investigation, Validation, Writing – Review & Editing; **Pandula Pallewatta:** Investigation, Validation; **Aboubakar Abdinur Hersi:** Data Curation, Visualization; **Shehan Maxwell Beruwalage:** Resources, Investigation; **Deshan Sachintha Kannangara:** Writing, Review and Editing.

7.2. Institutional Review Board Statement

Not applicable.

7.3. Informed Consent Statement

Not applicable.

7.4. Data Availability Statement

The datasets used in this study are publicly available benchmark datasets, namely ShanghaiTech, UCF-QNRF, and UCF-CC-50. ShanghaiTech Dataset (Parts A and B) is available at: https://www.kaggle.com/datasets/hosammhmdali/shanghai-tech-dataset-part-a-and-part-b. UCF-QNRF Dataset is available at: https://www.crcv.ucf.edu/data/ucf-qnrf/. UCF-CC-50 Dataset is available at: https://www.crcv.ucf.edu/data/ucf-cc-50/. All datasets are publicly available for research purposes and contain no personally identifiable information.

7.5. Acknowledgment

Not Applicable.

7.6. Conflicts of Interest

The authors declare no conflicts of interest.

8. 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 Recognit. Lett.*, vol. 107, pp. 3–16, May 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 *IEEE Conference on Computer Vision and Pattern Recognition,* 2013. https://doi.org/10.1109/CVPR.2013.329.

[3] A. B. Chan, Z. S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2008, https://doi.org/10.1109/CVPR.2008.4587569.

[4] A. B. Chan and N. Vasconcelos, "Counting people with low-level features and bayesian regression," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2160–2177, Apr. 2012, https://doi.org/10.1109/TIP.2011.2172800.

[5] V Lempitsky, A Zisserman, "Learning to count objects in images," *Advances in Neural Information Processing Systems 23 (NIPS 2010)*, 2010. https://proceedings.neurips.cc/paper/2010/hash/fe73f687e5bc5280214e0486b273a5f9-Abstract.html.

[6] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-Image Crowd Counting via Multi-Column Convolutional Neural Network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2016. https://doi.org/10.1109/CVPR.2016.70.

[7] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 4031–4039, Nov. 2017, https://doi.org/10.1109/CVPR.2017.429.

[8] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann, "DecideNet: Counting Varying Density Crowds Through Attention Guided Detection and Density Estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2018. https://doi.org/10.1109/CVPR.2018.00545.

[9] H. Idrees et al., "Composition Loss for Counting, Density Map Estimation and Localization in Dense Crowds," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 532-546, 2018. https://openaccess.thecvf.com/content_ECCV_2018/papers/Haroon_Idrees_Composition_Loss_for_ECCV_2018_paper.pdf.

[10] Z. Ma, X. Wei, X. Hong, and Y. Gong, "Bayesian Loss for Crowd Count Estimation With Point Supervision," 2019. Accessed: Jan. 18, 2026. [Online] Available: https://github.com/ZhihengCV/

[11] B. Wang, H. Liu, D. Samaras, and M. H. Nguyen, "Distribution Matching for Crowd Counting," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 1595–1607, 2020. https://proceedings.neurips.cc/paper/2020/hash/118bd558033a1016fcc82560c65cca5f-Abstract.html.

[12] VA. 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.

[13] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 116-131. https://openaccess.thecvf.com/content_ECCV_2018/html/Ningning_Light-weight_CNN_Architecture_ECCV_2018_paper.html.

[14] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More Features From Cheap Operations," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 2020. https://doi.org/10.1109/CVPR42600.2020.00165.

[15] 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.

[16] S. Zagoruyko and N. Komodakis, "Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer," *arXiv preprint arXiv:1612.03928*, 2016, https://doi.org/10.48550/arXiv.1612.03928.

[17] L. Liu, J. Chen, H. Wu, T. Chen, G. Li, and L. Lin, "Efficient Crowd Counting via Structured Knowledge Transfer," *MM 2020 - Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2645–2654, Oct. 2020, https://doi.org/10.1145/3394171.3413938.

[18] T. Furlanello, Z. C. Lipton, M. Tschannen, L. Itti, and A. Anandkumar, "Born Again Neural Networks," Jul. 03, 2018, PMLR. Accessed: Jan. 18, 2026. [Online]. Available: https://proceedings.mlr.press/v80/furlanello18a.html.

[19] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep Mutual Learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2018. https://doi.org/10.1109/CVPR.2018.00454.

[20] Y. Lee, J. Willette, J. Kim, J. Lee, and S. J. Hwang, "Exploring the Role of Mean Teachers in Self-supervised Masked Auto-Encoders," *arXiv preprint arXiv:2210.02077*, 2022. https://doi.org/10.48550/arXiv.2210.02077.

[21] I. S. Topkaya, H. Erdogan, and F. Porikli, "Counting people by clustering person detector outputs," *11th IEEE International Conference on Advanced Video and Signal-Based Surveillance*, AVSS 2014, pp. 313–318, Oct. 2014, https://doi.org/10.1109/AVSS.2014.6918687.

[22] Y. Li, X. Zhang, and D. Chen, "CSRNet: Dilated Convolutional Neural Networks for Understanding the Highly Congested Scenes," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2018. https://doi.org/10.1109/CVPR.2018.00120.

[23] X. Chen, Y. Bin, N. Sang, and C. Gao, "Scale pyramid network for crowd counting," *Proceedings - 2019 IEEE Winter Conference on Applications of Computer Vision, WACV 2019*, pp. 1941–1950, Mar. 2019, https://doi.org/10.1109/WACV.2019.00211.

[24] X. Jiang *et al.*, "Crowd Counting and Density Estimation by Trellis Encoder-Decoder Networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 2019. https://doi.org/10.1109/CVPR.2019.00629.

[25] L. Zhu et al., "Dual Path Multi-Scale Fusion Networks with Attention for Crowd Counting," *arXiv preprint arXiv:1902.01115,* 2019. https://doi.org/10.48550/arXiv.1902.01115.

[26] X. Shi, X. Li, C. Wu, S. Kong, J. Yang, and L. He, "A Real-Time Deep Network for Crowd Counting," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2020-May, pp. 2328–2332, May 2020, https://doi.org/10.1109/ICASSP40776.2020.9053780.

[27] J. Chen, S. Xiu, X. Chen, H. Guo, and X. Xie, "Flounder-Net: An efficient CNN for crowd counting by aerial photography," *Neurocomputing*, vol. 420, pp. 82–89, Jan. 2021, https://doi.org/10.1016/J.NEUCOM.2020.09.001.

[28] G. Gao, Q. Liu, Z. Hu, L. Li, Q. Wen, and Y. Wang, "PSGCNet: A Pyramidal Scale and Global Context Guided Network for Dense Object Counting in Remote-Sensing Images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022, https://doi.org/10.1109/TGRS.2022.3153946.

[29] X. Guo, K. Song, M. Gao, W. Zhai, Q. Li, and G. Jeon, "Crowd counting in smart city via lightweight Ghost Attention Pyramid Network," *Future Generation Computer Systems*, vol. 147, pp. 328–338, Oct. 2023, https://doi.org/10.1016/J.FUTURE.2023.05.013.

[30] Y. C. Li, R. S. Jia, Y. X. Hu, and H. M. Sun, "A lightweight dense crowd density estimation network for efficient compression models," *Expert Syst. Appl.*, vol. 238, p. 122069, Mar. 2024, https://doi.org/10.1016/J.ESWA.2023.122069.

[31] F. Zhu, H. Yan, X. Chen, and T. Li, "Real-time crowd counting via lightweight scale-aware network," *Neurocomputing*, vol. 472, pp. 54–67, Feb. 2022, https://doi.org/10.1016/J.NEUCOM.2021.11.099.

[32] G. Jiang, R. Wu, Z. Huo, C. Zhao, and J. Luo, "LigMSANet: Lightweight multi-scale adaptive convolutional neural network for dense crowd counting," *Expert Syst. Appl.*, vol. 197, p. 116662, Jul. 2022, https://doi.org/10.1016/J.ESWA.2022.116662.

[33] S. Wang, Z. Pu, Q. Li, and Y. Wang, "Estimating crowd density with edge intelligence based on lightweight convolutional neural networks," *Expert Syst. Appl.*, vol. 206, p. 117823, Nov. 2022, https://doi.org/10.1016/J.ESWA.2022.117823.

[34] Y. Chaudhuri, A. Kumar, O. C. Phukan, and A. B. Buduru, "A Lightweight Feature Fusion Architecture For Resource-Constrained Crowd Counting," *arXiv preprint arXiv:2401.05968,* 2024. https://doi.org/10.48550/arXiv.2401.05968.

[35] Y. Chen, H. Zhao, M. Gao, and M. Deng, "A Weakly Supervised Hybrid Lightweight Network for Efficient Crowd Counting," *Electronics 2024, Vol. 13, Page 723*, vol. 13, no. 4, p. 723, Feb. 2024, https://doi.org/10.3390/electronics13040723.

[36] Y. Li, F. Yu, and Q. Chen, "Lightweight Dynamic Convolutional Network for Crowd Counting Based on Curriculum Reinforcement Learning," *IEEE Transactions on Artificial Intelligence*, 2025, https://doi.org/10.1109/TAI.2025.3566923.

[37] Y. Liu, G. Cao, H. Shi, and Y. Hu, "Lw-Count: An Effective Lightweight Encoding-Decoding Crowd Counting Network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 10, pp. 6821–6834, Oct. 2022, https://doi.org/10.1109/TCSVT.2022.3171235.

[38] L. Liang, H. Zhao, F. Zhou, M. Ma, F. Yao, and X. Ji, "PDDNet: lightweight congested crowd counting via pyramid depth-wise dilated convolution," *Applied Intelligence 2022 53:9*, vol. 53, no. 9, pp. 10472–10484, Aug. 2022, https://doi.org/10.1007/S10489-022-03967-6.

[39] H. Lee and J. Lee, "TinyCount: an efficient crowd counting network for intelligent surveillance," *Journal of Real-Time Image Processing*, vol. 21, no. 4, pp. 153-, Aug. 2024, https://doi.org/10.1007/S11554-024-01531-8.

[40] Y. Liu, Q. Yi, and J. Zeng, "Reducing Capacity Gap in Knowledge Distillation with Review Mechanism for Crowd Counting," *arXiv preprint arXiv:2206.05475*, 2022. https://doi.org/10.48550/arXiv.2206.05475.

[41] M. Jiang, J. Lin, and Z. Jane Wang, "ShuffleCount: Task-Specific Knowledge Distillation for Crowd Counting," *Proceedings - International Conference on Image Processing, ICIP*, pp. 999–1003, 2021, https://doi.org/10.1109/ICIP42928.2021.9506698.

[42] R. Wang *et al.*, “Efficient Crowd Counting via Dual Knowledge Distillation,” *IEEE Transactions on Image Processing*, vol. 33, pp. 569–583, 2024, https://doi.org/10.1109/TIP.2023.3343609.

[43] M. A. Khan, H. Menouar, R. Hamila, and A. Abu-Dayya, “Crowd counting at the edge using weighted knowledge distillation,” *Scientific Reports 2025 15:1*, vol. 15, no. 1, pp. 11932-, Apr. 2025, https://doi.org/10.1038/s41598-025-90750-5.

[44] VW. Zhou, X. Yang, W. Yan, and Q. Jiang, “Hybrid Knowledge Distillation for RGB-T Crowd Density Estimation in Smart Surveillance Systems,” *IEEE Internet Things J.*, vol. 12, no. 7, pp. 9276–9289, 2025, https://doi.org/10.1109/JIOT.2024.3506624.

[45] W. Zhou, X. Yang, X. Dong, M. Fang, W. Yan, and T. Luo, “MJPNet-S∗: Multistyle Joint-Perception Network with Knowledge Distillation for Drone RGB-Thermal Crowd Density Estimation in Smart Cities,” *IEEE Internet Things J.*, vol. 11, no. 11, pp. 20327–20339, Jun. 2024, https://doi.org/10.1109/JIOT.2024.3369642.

[46] Y. Gu, “Perspective-aware distillation-based crowd counting,” *ACM International Conference Proceeding Series*, pp. 123–128, Jul. 2020, https://doi.org/10.1145/3417188.3417195.

[47] S. Yun, J. Park, K. Lee, and J. Shin, “Regularizing Class-Wise Predictions via Self-Knowledge Distillation,” 2020. Accessed: Nov. 28, 2025. [Online]. Available: https://github.com/alinlab/cs-kd.

[48] Z. Allen-Zhu and Y. Li, “Towards Understanding Ensemble, Knowledge Distillation and Self-Distillation in Deep Learning,” *arXiv preprint arXiv:2012.09816*, 2020. https://doi.org/10.48550/arXiv.2012.09816.

[49] Y. Hou, Z. Ma, C. Liu, and C. C. Loy, “Learning Lightweight Lane Detection CNNs by Self Attention Distillation,” *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1013–1021, Aug. 2019, https://doi.org/10.1109/ICCV.2019.00110.

[50] VM. Ji, S. Shin, S. Hwang, G. Park, and I.-C. Moon, “Refine Myself by Teaching Myself: Feature Refinement via Self-Knowledge Distillation,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. https://doi.org/10.1109/CVPR46437.2021.01052.

[51] K. Kim, B. Ji, D. Yoon, and S. Hwang, “Self-Knowledge Distillation With Progressive Refinement of Targets,” in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. https://doi.org/10.1109/ICCV48922.2021.00650.

[52] Z. Yang, A. Zeng, Z. Li, T. Zhang, C. Yuan, and Y. Li, “From Knowledge Distillation to Self-Knowledge Distillation: A Unified Approach with Normalized Loss and Customized Soft Labels,” in *IEEE/CVF International Conference on Computer Vision (ICCV),* 2023. https://doi.org/10.1109/ICCV51070.2023.01576.

[53] J. Yi, F. Chen, Z. Shen, Y. Xiang, S. Xiao, and W. Zhou, “An Effective Lightweight Crowd Counting Method Based on an Encoder-Decoder Network for Internet of Video Things,” *IEEE Internet Things J.*, vol. 11, no. 2, pp. 3082–3094, Jan. 2024, https://doi.org/10.1109/JIOT.2023.3294727.

[54] VM. Xi and H. Yan, “Lightweight multi-scale network with attention for accurate and efficient crowd counting,” *The Visual Computer*, vol. 40, no. 6, pp. 4553–4566, Sep. 2023, https://doi.org/10.1007/S00371-023-03099-Z.