ORIGINAL RESEARCH PAPER

LCDnet: a lightweight crowd density estimation model for real-time video surveillance

Muhammad Asif Khan[1] · Hamid Menouar[1] · Ridha Hamila[2]

Abstract

Automatic crowd counting using density estimation has gained significant attention in computer vision research. As a result, a large number of crowd counting and density estimation models using convolution neural networks (CNN) have been published in the last few years. These models have achieved good accuracy over benchmark datasets. However, attempts to improve the accuracy often lead to higher complexity in these models. In real-time video surveillance applications using drones with limited computing resources, deep models incur intolerable higher inference delay. In this paper, we propose (i) a Lightweight Crowd Density estimation model (LCDnet) for real-time video surveillance, and (ii) an improved training method using curriculum learning (CL). LCDnet is trained using CL and evaluated over two benchmark datasets i.e., DroneRGBT and CARPK. Results are compared with existing crowd models. Our evaluation shows that the LCDnet achieves a reasonably good accuracy while significantly reducing the inference time and memory requirement and thus can be deployed over edge devices with very limited computing resources.

1 Introduction

Crowd counting is an interesting research area that involves computer vision and deep learning to estimate the number of people in an images or video frames. Recently, it has gained significant attention in the computer vision community due to the significance of the problem. Crowd counting is generally implemented in two ways: (i) counting objects (input is an image and the output is a number i.e., total head count in the image.), and (ii) density map estimation (input is an image and the output is the density map of the crowd which is then integrated to get the total head count.). Traditional methods for crowd counting were all based on

detecting hand-crafted local features such as full body [1, 2], body parts [3–5], shapes [6], or global features such as foreground [7], edge [8], texture [9] and gradient features [10, 11] and then use machine learning models such as linear regression, ridge regression, Gaussian process, support vector machines (SVMs), random forest, gradient boost, and neural networks to provide the total count or a density map of the image. However, all the accuracy of these methods significantly degrades on images with dense crowds due to challenges such as occlusions, low resolution, foreshortening and perspectives.

Recent research on crowd counting shows the efficacy of deep learning methods for crowd counting [12]. Convolution neural networks (CNNs) due to their strong capability of auto feature extraction [13, 14]. Although even small CNN models [15] outperform traditional counting methods, their accuracy degrade on high density scenes.

To achieve higher accuracy in dense scenes, deeper models with a large size of parameters are developed [16–19]. These deep models although achieve good accuracy create performance bottlenecks in real-time applications due to large memory requirement, higher training complexity, and large inference delay. In contrast, small-sized CNN models offer several benefits in real-time video surveillance

✉ Muhammad Asif Khan
  mkhan@qu.edu.qa

  Hamid Menouar
  hamidm@qmic.com

  Ridha Hamila
  hamila@qu.edu.qa

[1] Qatar Mobility Innovations Center (QMIC), Qatar University, Doha, Qatar

[2] Electrical Engineering, Qatar University, Doha, Qatar

e.g., they incur low inference delay, require low memory for deployment on embedded devices, quickly update over-the-air, and can be trained, fine-tuned and run in distributed manner [20]. However, lightweight and shallow CNN models are usually disregarded by some due to their limited accuracy. Contrary to that, we believe that leveraging best practices such as carefully designing the model architecture, the use of accurately annotated training data, and efficient learning strategies can jointly improve the accuracy of shallow models to a greater extent [21].

1.1 Model design strategy

Generally a good choice of convolution filters play a crucial role in feature learning and contribute to reducing the size of the model. In essence, larger filters (5×5) are more expensive and thus should be replaced by smaller filters (3×3). As an example, a stack of three (3×3) convolution layers is preferred over a single convolution layer of larger receptive fields such as 7×7 or 9×9 because of the non-linear activations between them. It also has less number of parameters ($3 \times 3^2 \times C^2 < 1 \times 7^2 \times C^2$ i.e., 81% less parameters) and are computed faster. Further dimension reduction can be applied using $a \times 1$ convolution before expensive convolutions (e.g., 5×5 or 3×3) [22]. Furthermore, spatially separable convolutions are preferred i.e., a 3×3 convolution can be decomposed in two sequential 1×3 and 3×1 convolutions which leads to the same number of parameters and can even achieve better learning.

1.2 Data annotation strategy

In crowd density estimation, the point annotations on top of the heads in crowd image form a sparse "dot-map" or "localization-map" which is converted into a density map by convolving the head position with a Gaussian kernel. The scale parameter in the Gaussian kernel visually creates a blob around the point (pixel). A good density map is more accurate if the blob size covers the entire head and do not non-overlap with neighboring heads. However, due to the camera perspective effect, the size of heads vary in the same image. Adaptive kernels are used to select different values of the scale parameter which solve this problem to some extent. To cope with the perspective distortion and scale variations in crowd images, recent works propose often very deep models with complex architectures. However, unlike images captured with CCTV camera, in aerial images captured from drones, the perspective distortion is minimum and the scale variation is directly related to the drone altitude. If the drone-altitude is known, it can be used to generate accurate density maps.

1.3 Model learning strategy

The idea of curriculum learning (CL) in neural network was first presented in [23]. The idea behind CL is the natural learning process in humans and animals i.e., humans learn better when concepts are presented in specific order of complexity i.e., from simpler to difficult tasks. Unlike traditional learning methods, in curriculum learning the training samples are sorted by order of (typically increasing) complexity. CL has been demonstrated as an effective strategy to improve the learning capability and faster convergence in various tasks e.g., computer vision [24–26], natural language processing (NLP) [27, 28], reinforcement learning [29–31] etc. Recent studies [32, 33] inspire to adopt CL in our research.

This work leverages the aforementioned three strategies for generating ground truth density maps and to design and train an extremely lightweight crowd density estimation model. In the first step, we designed the model (LCDnet) by carefully choosing convolution filters of different sizes in different layers. To keep the model more compact, we used less number of filters in the initial layers so even with the larger input feature maps, the computational load is controlled. To further alleviate the computational complexity, we used rectangular filters of size (1×3) and (3×1) instead of square filters. This also improves the learning performance as indicated in [22]. The resultant CNN model is a shallow network with only 0.05 Million parameters. Next, to train the shallow model with drone-captured aerial images, we generated high quality (accurate) density maps. We tested different scale values of the Gaussian function and empirically found the most accurate values for each image. Lastly, the curriculum learning technique is employed to improve the learning performance of the model.

The contribution of our work is as follows:

- A lightweight CNN model (LCDnet) with fewer parameters, low memory requirement, and faster run-time than existing models.
- Generate density maps from sparse localization maps by considering drone-altitude to create adaptive Gaussian kernels to improve learning.
- Propose an efficient strategy based on curriculum-learning approach to further improve model training and convergence.
- Experimental demonstration of reasonably good performance using LCDnet on benchmark datasets. The accuracy is comparable to existing models of double-size than LCDnet.

2 Related work

A number of crowd counting datasets exists each of which can be divided into three categories. *Surveillance-view* datasets containing indoor or outdoor images collected by surveillance cameras (e.g., Mall [34], UCSD [35], WorldExpo'10 [15], ShanghaiTech Part B [36], *Free-view dataset* containing images from different sources including Internet (e.g., UCF-CC-50 [37], UCF-QNRF [38], ShanghaiTech Part A [36]), and *Drone view datasets* containing images collected using drones (e.g., DroneRGBT [39], CARPK [40]). Most of the earlier research works on crowd counting have been using the datasets in the first two categories. The drone-view datasets have been available recently.

While there has been different approaches for crowd counting, density estimation using CNN is the most widely used method. The first known CNN model for density estimation and counting is CrowdCNN [15]. The CrowdCNN model consists of three convolution layers followed by three fully connected layers. Following this, numerous works proposed different CNN-based models for crowd counting.

A multi-column CNN (MCNN) model is proposed in [36] which consists of three CNN columns, each containing filters of receptive fields of different sizes (small, medium large). The outputs of the three columns are combined to predict the final density map. MCNN exhibits good performance to adapt to the scale variations in images due to perspective effects or different image resolutions. A two column CNN model (CrowdNet) is proposed in [41]. The model consists of two CNN columns i.e., a deep network (five CNN layers) and a shallow network (three CNN layers). The outputs of both networks are combined to predict the final density map. A Switched CNN (SCNN) is proposed in [42], which consists of two parts, a Switch and a CNN regressor. The CNN regressor consists of three independent columns each with different size receptive fields. Patches from the input image are first fed to the Switch network, which relay it to one of the CNN regressor to predict the density map. The intuition in SCNN is to build CNN model which can adapt to the large scale variations without increasing the model computational complexity i.e., a patch is passed through only one column in the regressor. This reduces the computational complexity when compared to other multi-column CNN models e.g., MCNN and CrowdNet.

Unlike multi-column CNN models, authors in [43] proposed a single column network called multi scale CNN (MSCNN) to learn the scale variations. MSCNN uses three Inception modules [22] called multi-scale blobs (MSBs). Each MSB consists of multiple filters with different kernel size and is able to extract scale-relevant features. The aforementioned CNN models can adapt to the scale variations introduced in the training data but may fail to generalize well [18]. A cascaded multi-task learning (CMTL) model [18] is proposed to adapt to the wide variations of density levels in images. CMTL is also a two column network. The first column is a high-level prior that classify an input image into groups based on the total count in the image. The features learned by the high level prior are shared with the second column that estimates the respective density map.

The previous models mostly used multi-column architectures to learn scale-relevant features in crowd images achieves good results. To further improve the counting accuracy in highly congested scenes, authors in [16] propose deeper architecture by utilizing transfer learning. The Congested Scene Recognition (CSRNet) model [16] uses VGG16 [44] (first 10 layers) as the front-end to extract features, and a back-end network with dilated convolution to substitute the pooling layers thus avoiding the loss of spatial information. Transfer learning an improve the feature extraction capability of the crowd counting model and has been recently adopted in CANNet [45], GSP [46], TEDnet [47], Deepcount [48], SASNet [49], M-SFANet [19], and SGANet [50].

The aforementioned models often produces output density maps of lower resolution than the input image and uses patch-based training. The Trellis Encoder–Decoder Network (TEDnet) is proposed [47] which uses whole image as the model input and preserve the size of the density map to the actual resolution. Another encoder-decoder model with special Inception [22]-like modules is scale aggregation network (SANet) proposed in [17]. Like TEDnet, SANet also produces high resolution density map but has a simpler model structure.

Our proposed model LCDnet simultaneously achieves two benefits as compared to the aforementioned models. First, it is extremely lightweight as compared to other models and can run faster even at edge devices with limited compute resources (a comparison is shown later in the paper). Secondly, it also produces density maps of better quality of size $(\frac{1}{2})$ of input size as compared to other methods e.g., CSRNet $(\frac{1}{8})$ and MCNN $(\frac{1}{4})$. It also requires least amount of memory to fit in on-chip caches.

3 Proposed method

The aforementioned CNN-based crowd models are designed to improve counting accuracy. However, in a typical crowd monitoring system, the user may be interested in a rough estimation of crowd densities (e.g., low, medium, high, very high etc.) located in a geographical area rather than the exact

count. Our goal in this paper is to develop a very lightweight model which can reasonably detect the presence of crowd scenes to an extent useful in analyzing crowds but essentially run faster on edge devices with limited computing resource. To this end, we propose a crowd density estimation model (LCDnet).

3.1 Network architecture

The architecture of LCDnet is shown in Fig. 1. It consists of six convolution (Conv) layers. Conv1 consists of 64 (5×5) filters. The output of Conv1 layer is fed to Conv2 and Conv3, each having 32 (3×3) filters. The outputs from Conv2 and Conv3 is fed to Conv4 and Conv5 both has 64 (3×3) filters respectively. The outputs of Conv4 and Conv5 is concatenated and fed to Conv6 which consists of 128 (1×1) filters to predict the final density map. It is worthy to note that LCDnet returns the density map rather than the crowd density class. There are two benefits for this. First, the density map preserves the location information of crowd which can be used to localize the crowd in the real-world. Second, it is easy to determine the total count (whole scene) or local count (specific part of the scene) from the predicted density map and even use the count information to configure user-defined crowd densities based on the application requirement.

In the proposed LCDnet architecture, the first convolution layer is used to detect features such as edges. These detected features are then used in two columns. Both columns contain three layers i.e., two layers of 32 filters of size 1×3, and 3×1 (in reverse order in column 2), respectively, which is

followed by a layer of 64 filters of 3×3 size. The outputs of both columns are concatenated and fed to a 1×1 conv layer of size 128, which generates a density map. The output density map is half size of the input image.

3.2 Ground truth generation

If x_i is a pixel containing the head position, it can be represented by a delta function $\delta(x - x_i)$. The density map is generated by convolving the delta function with a Gaussian kernel G_σ.

$$Y = \sum_{i=1}^{N} \delta(x - x_i) * G_\sigma, \tag{1}$$

where, N is the total number of annotated points (i.e., total count of heads) in the image. The integral of density map Y is equal to the total head count in the image. Visually, this operation creates a blurring of each head annotation using the scale parameter σ. There are various kernel settings to generate a variety of density maps. The most basic approach is to keep σ fixed value, which means that the density map will apply same kernel to all head positions irrespective of their scale in the image [17]. As head sizes in image can vary due to camera prospective, a single value of σ may not be a good choice. Hence, some recent works propose to use adaptive Gaussian kernels to create density maps. The value of σ is calculated as the average distance to k-nearest neighboring head annotations. Visually, it generates lower degree of Gaussian blur for dense crowds and higher degree for region of sparse density in crowd scene. Typical settings includes

Fig. 1 LCDnet with curriculum learning
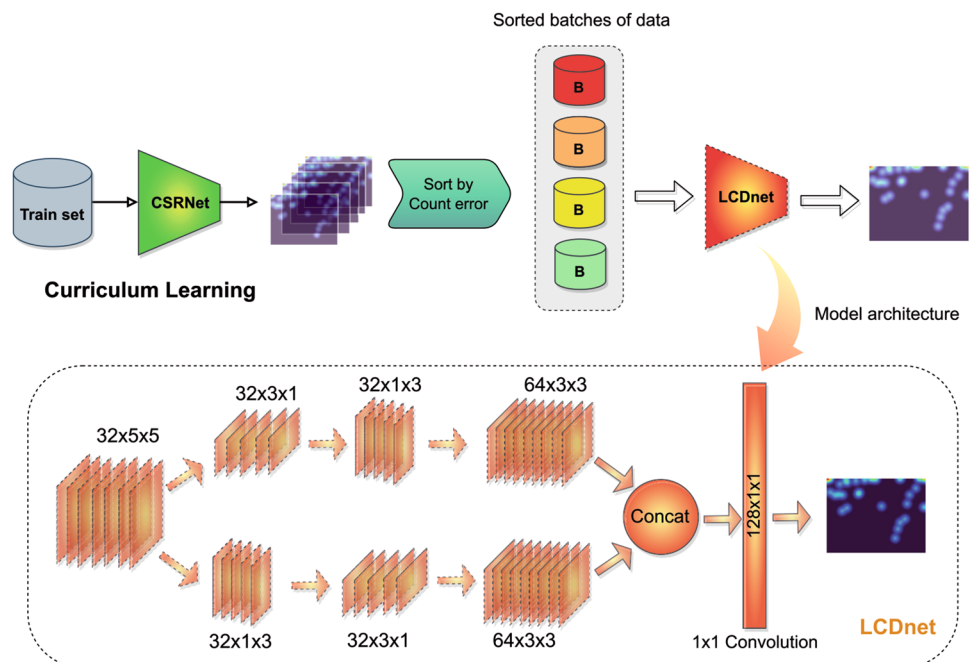
$k = 1$ [38], $k = 10$ [17, 43]. Although adaptive Gaussian kernel may produce better results on images with large scale variations, our intuition is that drone images typically have less scale variations as compared to surveillance images e.g., from CCTV. The scale variations in drone images result from drone flying altitudes which do not vary too much due to regulatory measures. Thus, the scale variations are limited and a single value of σ can be experimentally determined to produce density maps. In our experiments, we empirically determined the value of σ based on the drone altitudes. The datasets used in this study contain images captured from different altitudes. Thus, we first segregated images into different groups by drone altitudes and empirically found separate values of σ for each group.

3.3 Training

The image resolutions in the datasets used in this study are not very high, thus we use whole image-based training operations without extracting patches or downsampling operations on training images. However, to avoid model overfitting, data augmentations techniques such as horizontal flipping, and random brightness and contrast are applied. The kernels in all Conv layers are randomly initialized using Gaussian distribution with the value of standard deviations 0.01. We used Adam optimizer with a base learning rate 0.0001. The loss function used is pixel-wise euclidean distance between the target and predicted density maps which is defined in Eq. (2).

$$L(\Theta) = \frac{1}{N} \sum_{1}^{N} ||D(X_i;\Theta) - D_i^{gt}||_2^2, \tag{2}$$

where N is the number of samples in training data, $D(X_i;\Theta)$ is the predicted density map with parameters Θ for the input image X_i, and D_i^{gt} is the ground truth density map.

We further applied curriculum learning technique to improve the learning performance of our model. In CL settings, we used transfer learning using CSRNet to determine the difficulty level of each image. Based on the counting error, images are sorted in ascending order before packing them into mini-batches. Thus, mini-batches are created in the order of their cumulative complexity.

4 Experiments and results

The proposed model (LCDnet) was trained on a single GPU (Nvidia RTX-8000) using PyTorch deep learning framework. We also implemented and trained other models used in this study from the scratch for fair comparison.
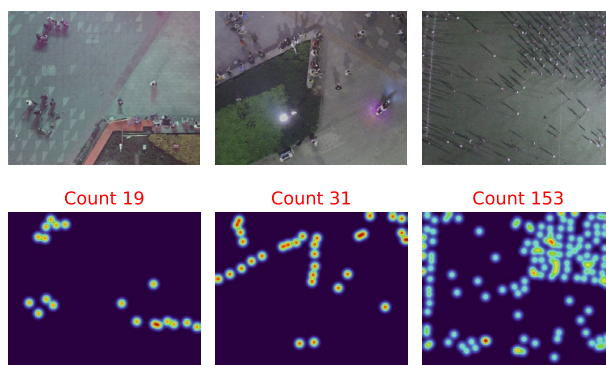

Fig. 2 Sample images (top) and their corresponding density maps (bottom) from DroneRGBT dataset
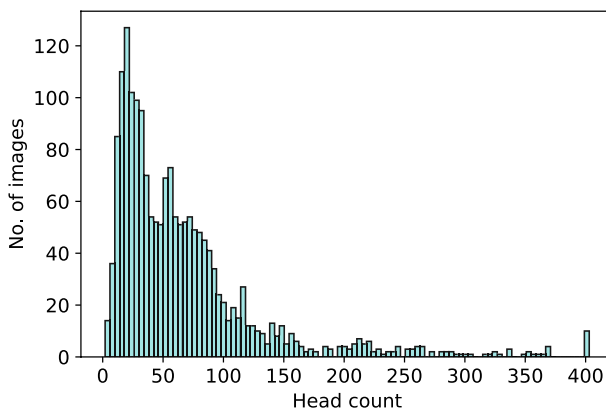

Fig. 3 Count distribution in DroneRGBT dataset

4.1 Datasets

We evaluate the proposed model on two benchmark datasets i.e., DroneRGBT and CARPK. The DroneRGBT dataset contains images of people whereas the CARPK dataset contains images of cars, both captured from drones.

4.1.1 DroneRGBT

The dataset contains 3600 RGB and thermal image pairs with a spatial resolution of 512×640 pixels. The images cover a wide range of scenes e.g., campus, streets, parks, parking lots, playgrounds, and plazas. The dataset is divided into training set (1807 samples) and test set (912 samples) in such a way that both the training and test set include diverse images (i.e., different scenes, crowd densities, illumination, and scales) to avoid overfitting. The dataset provides head annotations of people. The count distribution and sample images from the dataset are presented in Fig. 3 and Fig. 2, respectively.
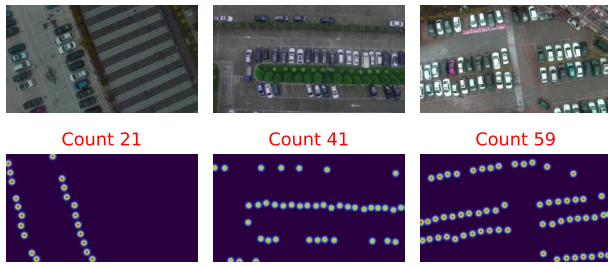
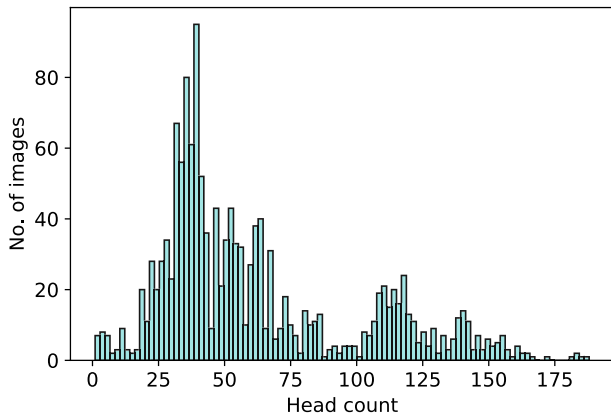**Fig. 4** Sample images (top) and their corresponding density maps (bottom) from CARPK dataset

Fig. 5 Count distribution in CARPK dataset

4.1.2 CARPK

The dataset contains 1 images of cars from 4 different parking lots captured with a drone. The dataset is divided into a training set containing 989 images and a test set containing 459 images. The dataset has a total number of 89, 777 cars. The original dataset contains bounding box annotations, however we transformed the original annotations to dot annotation by taking the center of the bounding box. The count distribution and sample images from the dataset are presented in Figs. 5 and 4, respectively.

4.2 Evaluation metrics

Most of the existing works on crowd counting use mean absolute error (MAE) and Grid average mean absolute error (GAME) to evaluate the accuracy of the model. MAE is the average of absolute error in predicted counts and actual counts of all images. It is calculated using 3:

$$\text{MAE} = \frac{1}{N} \sum_{1}^{N} (e_n - \hat{g}_n), \tag{3}$$

where, N is the total number of images in the dataset, g_n is the ground truth (actual count) and $\hat{e}_n$ is the prediction (estimated count) in the nth image. While MAE is the most widely used metric in crowd counting research and is often used to compare various models, MAE provide image-wide counting and does not provide where the estimations have been done in the image. Owing to the possible estimation errors in MAE, authors in [51] proposed Grid Average Mean absolute Error (GAME). In GAME, an image is divided into 4^L non-overlapping patches and compute MAE separately for each patch. Thus, GAME poses a more robust and accurate estimation for crowd counting applications. It is defined in Eq. 4.

$$\text{GAME} = \frac{1}{N} \sum_{n=1}^{N} (\sum_{l=1}^{4^L} |e_n^l - g_n^l|) \tag{4}$$

The GAME metric is more robust to localisation errors in density estimation by calculating localized error among the target and predicted density maps. We set the value of $L = 4$, thus each density map is divided into a grid size of 4×4 creating 16 patches. The absolute difference in the head count for each patch is measured and summed for all patches of the same density map, then averaged over the whole dataset. We compared the LCDnet model against existing models over the two metrics to provide a fair evaluation of the model accuracy. However, the true benefit of LCDnet is the lower model complexity at the cost of tolerable counting error. In addition, the performance is measured over two other metrics i.e., structural similarity index (SSIM), peak signal-to-noise ratio (PSNR). Both SSIM and PSNR evaluate the quality of the predicted density maps and are measured in Eq. (5) and (6) as follows:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_x\sigma_y C_2)}{(\mu_z^2\mu_y^2 + C_1)(\mu_z^2\mu_y^2 + C_2)}, \tag{5}$$

where $\mu_x, \mu_y, \sigma_x, \sigma_y$ represents the means and standard deviations of the actual and predicted density maps, respectively.

$$\text{PSNR} = 10log_{10}\left(\frac{\text{Max}(I^2)}{\text{MSE}}\right), \tag{6}$$

where $\text{Max}(I^2)$ the maximal in the image data. If it is an 8-bit unsigned integer data type, the $\text{Max}(I^2) = 255$.

4.3 Evaluation results

We compared the proposed model (LCDnet) against two mainstream crowd density estimation models i.e., MCNN [36], and CSRNet [16]. MCNN is a relatively small-sized CNN model which has gained good counting accuracy over several benchmark datasets as compared to other models of similar size. CSRNet on the other hand is a deep CNN model

that uses VGG-16 [44] as a front-end and has shown high accuracy in dense crowd scenes. Both models have been used for comparison in many crowd counting studies and thus we chose them as candidate models for small-sized and large-sized models in this study.

We compare the performance of LCDnet in terms of counting accuracy at the cost of model size and complexity against the MCNN and CSRNet on DroneRGBT and CARPK datasets. While the primary comparison is done against MCNN and CSRNet, we additionally provide complexity comparison against some other well-known counting models to highlight the benefit of the proposed model (LCDnet). The model complexity comparison is shown in Table 1 whereas the accuracy comparison is depicted in Table 2. The inference time is computed on GPU server (Nvidia RTX 8GB), and two different edge devices (Nvidia Jetson Xavier and Jetson Nano). The system details of these devices are as follows:

- **Server:** GPU (Nvidia Quadro RTX-8000).
- **Jetson Xavior NX:** 64bit system with Processor (6-core NVIDIA Carmel ARM), Memory (8GB), GPU (NVIDIA Volta architecture with 384 NVIDIA CUDA cores and 48 Tensor cores).
- **Jetson Nano:** 64bit system with Quad-Core Arm Cortex-A57 MPCore, Memory (4GB), GPU (128-core NVIDIA Maxwell GPU).

On DroneRGBT dataset, LCDnet achieves MAE of 21.4 which is comparable with that of MCNN (17.9). In terms of complexity, LCDnet has almost half the number of parameters and half number of multiply-add-calculations (GMACs). LCDnet also incurs much lower ($\frac{1}{2}$×) inference delay as compared to MCNN. Although the accuracy of CSRNet is much higher than both LCDnet (3×) and MCNN (2.2×), it has a very huge size requiring large memory size and much higher (20×) inference delay than LCDnet.

On CARPK dataset, LCDnet achieves better results. It achieves MAE 13.1, which is close to MCNN (10.1) and slightly less than CSRNet (6.12). Some sample predictions using MCNN, CSRNet and the proposed LCDnet models over DroneRGBT and CARPK datasets, respectively, are shown in Figs. 6 and 7. It can be visualized that LCDnet has good detection capability and produces better quality density maps than MCNN for DroneRGBT dataset. We believe this is due to the use of small sized filters (1×3 and 3×1). The better quality of density map is expected and evident from the higher values of SSIM and PSNR.

5 Conclusion

This paper proposes a lightweight crowd density estimation model (LCDnet) for deployment over resource-constrained embedded devices (e.g., drones) suitable for real-time applications (e.g., surveillance) scenarios. The paper outlines various design principles and best practices used to develop efficient CNN architectures. LCDnet is designed by adopting three efficient strategies; (i) compact CNN model (ii) improved ground truth generation from head annotations and drone altitudes, and (iii) improved

Table 1 Comparison of proposed scheme (LCDnet trained with curriculum learning) against SOTA models for number of parameters (in Million), GMACs, size (in MB), and inference time (in milliseconds) for fixed input size

Model	Output	Parameters (M)	Size (MB)	GMACs	Inference time (s)		
					Server	Jetson Xavier	Jetson Nano
CrowdCNN [15]	1/4	1.66	6.65	3.96	0.071	0.073	0.23
MCNN [36]	1/4	0.13	0.53	8.82	0.05	0.10	0.21
CMTL [18]	1/4	2.45	9.82	39.82	0.098	0.31	0.62
CSRNet [16]	1/8	16.26	65.05	135.4	0.19	1.01	1.88
SANet [17]	1/4	0.25	1.02	8.97	0.075	0.12	0.23
LCDnet (ours)	1/2	0.05	0.21	4.85	0.006	0.05	0.10

Table 2 Accuracy comparison of the proposed scheme (LCDnet trained with curriculum learning) against SOTA models over DroneRGBT dataset [39] and CARPK dataset [40]

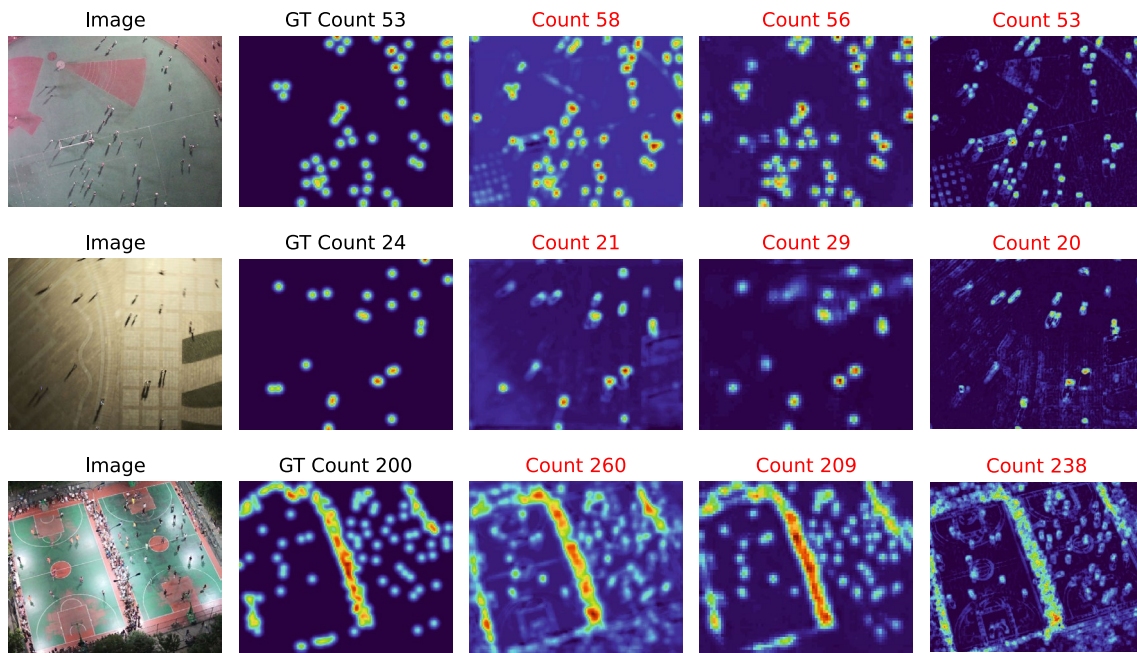| Method	DroneRGBT				CARPK			
	MAE	GAME	SSIM	PSNR	MAE	GAME	SSIM	PSNR
CrowdCNN [15]	26.6	48.2	0.52	17.3	15.6	49.1	0.63	18.8
MCNN [36]	17.9	44.49	0.54	18.2	10.3	42.40	0.76	19.21
CMTL [18]	18.1	40.5	0.53	17.1	10.2	41.3	0.73	19.0
CSRNet [16]	7.6	25.7	0.72	21.70	6.12	21.8	0.82	20.52
SANet [17]	16.2	34.5	0.59	19.4	9.8	27.2	0.76	19.8
LCDnet (ours)	21.4	46.9	0.60	21.39	13.1	45.2	0.79	20.14

Fig. 6 Comparison of predictions on on DroneRGBT dataset. The first two columns shows crowd images and their corresponding ground truth. Columns 3–4 shows predictions using MCNN [36] and CSRNet [16] without curriculum learning. Column 5 shows predictions using LCDnet (ours), respectively
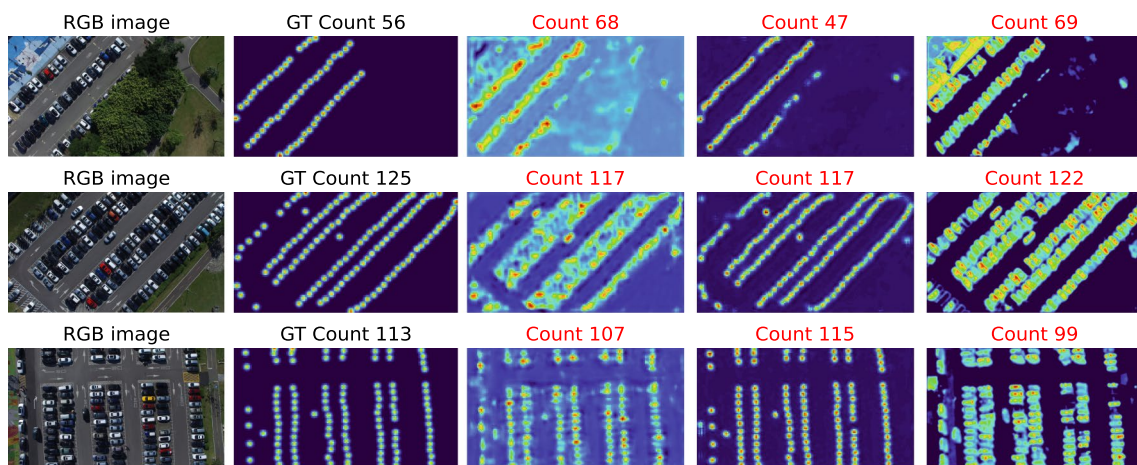

Fig. 7 Comparison of predictions on on CARPK dataset. The first two columns shows crowd images and their corresponding ground truth. Columns 3–4 shows predictions using MCNN [36] and CSR-Net [16]. without curriculum learning. Column 5 shows predictions using LCDnet (ours), respectively

training mechanism using curriculum learning. LCDnet is evaluated on two different datasets of drone-captured images i.e., DroneRGBT, CARPK. Our experimental analysis shows that LCDnet achieves reasonably good accuracy at much lower computational cost. The small memory footprint and lower inference time makes LCDnet a good fit for drone-based video surveillance.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Topkaya, I. S., Erdogan, H., Porikli, F.: Counting people by clustering person detector outputs. In: 2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 313–318 (2014). https://doi.org/10.1109/AVSS.2014.6918687

2. Tuzel, O., Porikli, F., Meer, P.: Pedestrian detection via classification on riemannian manifolds. IEEE Trans. Pattern Anal. Mach. Intell. **30**(10), 1713–1727 (2008). https://doi.org/10.1109/TPAMI.2008.75

3. Li, M., Zhang, Z., Huang, K., Tan, T.: Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In: 2008 19th International Conference on Pattern Recognition, pp. 1–4 (2008). https://doi.org/10.1109/ICPR.2008.4761705

4. Viola, P., Jones, M.: Robust real-time face detection. In: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, vol. 2, pp. 747–747 (2001). https://doi.org/10.1109/ICCV.2001.937709

5. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010). https://doi.org/10.1109/TPAMI.2009.167

6. Lin, Z., Davis, L.S.: Shape-based human detection and segmentation via hierarchical part-template matching. IEEE Trans. Pattern Anal. Mach. Intell. **32**(4), 604–618 (2010). https://doi.org/10.1109/TPAMI.2009.204

7. Davies, A.C., Yin, J.H., Velastín, S.A.: Crowd monitoring using image processing. Electron. Commun. Eng. J. **7**, 37–47 (1995)

8. Wu, B., Nevatia, R.: Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors. Int. J. Comput. Vis. **75**, 247–266 (2006)

9. Chen, K., Loy, C. C., Gong, S., Xiang, T.: Feature mining for localised crowd counting. In: BMVC (2012)

10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 886–893 (2005). https://doi.org/10.1109/CVPR.2005.177

11. Tian, Y., Sigal, L., Badino, H., la Torre, F.D., Liu, Y.: Latent gaussian mixture regression for human pose estimation. In: ACCV (2010)

12. Khan, M.A., Menouar, H., Hamila, R.: Revisiting crowd counting: State-of-the-art, trends, and future perspectives. Image Vis. Comput. **129**, 104597 (2023). https://doi.org/10.1016/j.imavis.2022.104597

13. Chu, J., Guo, Z., Leng, L.: Object detection based on multilayer convolution feature fusion and online hard example mining. IEEE Access **6**, 19959–19967 (2018). https://doi.org/10.1109/ACCESS.2018.2815149

14. Zhang, Y., Chu, J., Leng, L., Miao, J.: Mask-refined r-cnn: a network for refining object details in instance segmentation. Sensors (Basel, Switzerland) **20**, 1010 (2020)

15. Zhang, C., Li, H., Wang, X., Yang, X.: Cross-scene crowd counting via deep convolutional neural networks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 833–841 (2015)

16. Li, Y., Zhang, X., Chen, D.: Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1091–1100 (2018)

17. Cao, X., Wang, Z., Zhao, Y., Su, F.: Scale aggregation network for accurate and efficient crowd counting. In: ECCV (2018)

18. Sindagi, V.A., Patel, V.M.: Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6 (2017)

19. Thanasutives, P., ichi Fukui, K., Numao, M., Kijsirikul, B.: Encoder-decoder based convolutional neural networks with multi-scale-aware modules for crowd counting. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 2382–2389 (2021)

20. Iandola, F.N., Moskewicz, M.W., Ashraf, K., Han, S., Dally, W.J., Keutzer, K.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and<1 mb model size. arXiv:abs/1602.07360 (2016)

21. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.A.: Striving for simplicity: the all convolutional net. CoRR **abs/1412.6806** (2015)

22. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9 (2015)

23. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09, p. 41-48. Association for Computing Machinery, New York (2009). https://doi.org/10.1145/1553374.1553380.

24. Hacohen, G., Weinshall, D.: On the power of curriculum learning in training deep networks. ArXiv **abs/1904.03626** (2019)

25. Guo, S., Huang, W., Zhang, H., Zhuang, C., Dong, D., Scott, M.R., Huang, D.: Curriculumnet: Weakly supervised learning from large-scale web images. arXiv:abs/1808.01097 (2018)

26. Jiang, L., Meng, D., Mitamura, T., Hauptmann, A.: Easy samples first: self-paced reranking for zero-example multimedia search. In: Proceedings of the 22nd ACM International Conference on Multimedia (2014)

27. Platanios, E.A., Stretcu, O., Neubig, G., Póczos, B., Mitchell, T.M.: Competence-based curriculum learning for neural machine translation. arXiv:abs/1903.09848 (2019)

28. Tay, Y., Wang, S., Luu, A.T., Fu, J., Phan, M.C., Yuan, X., Rao, J., Hui, S.C., Zhang, A.: Simple and effective curriculum pointer-generator networks for reading comprehension over long narratives. In: ACL (2019)

29. Florensa, C., Held, D., Wulfmeier, M., Zhang, M., Abbeel, P.: Reverse curriculum generation for reinforcement learning. In: CoRL (2017)

30. Narvekar, S., Sinapov, J., Stone, P.: Autonomous task sequencing for customized curriculum design in reinforcement learning. In: IJCAI (2017)

31. Ren, Z., Dong, D., Li, H., Chen, C.: Self-paced prioritized curriculum learning with coverage penalty in deep reinforcement learning. IEEE Trans. Neural Netw. Learn. Syst. **29**, 2216–2226 (2018)

32. Li, W., Cao, Z., Wang, Q., Chen, S., Feng, R.: Learning error-driven curriculum for crowd counting. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 843–849 (2021)

33. Wang, Q., Lin, W., Gao, J., Li, X.: Density-aware curriculum learning for crowd counting. IEEE Trans. Cybern. **52**, 4675–4687 (2022)

34. Chen, K., Loy, C.C., Gong, S., Xiang, T.: Feature mining for localised crowd counting. In: Proceedings of the British Machine Vision Conference, pp. 21.1–21.11. BMVA Press (2012)

35. Chan, A.B., Liang, Z.S.J., Vasconcelos, N.: Privacy preserving crowd monitoring: counting people without people models or tracking. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7 (2008)

36. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: Single-image crowd counting via multi-column convolutional neural network. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 589–597 (2016). https://doi.org/10.1109/CVPR.2016.70

37. Idrees, H., Saleemi, I., Seibert, C., Shah, M.: Multi-source multi-scale counting in extremely dense crowd images. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2547–2554 (2013)

38. Idrees, H., Tayyab, M., Athrey, K., Zhang, D., Al-Maadeed, S.A., Rajpoot, N.M., Shah, M.: Composition loss for counting, density map estimation and localization in dense crowds. arXiv:abs/1808.01050 (2018)

39. Peng, T., Li, Q., Zhu, P.: Rgb-t crowd counting from drone: A benchmark and mmccn network. In: Computer Vision—ACCV 2020: 15th Asian Conference on Computer Vision. Kyoto, Japan, November 30–December 4, 2020, Revised Selected Papers, Part VI, pp. 497–513. Springer, Berlin (2020)

40. Hsieh, M.R., Lin, Y.L., Hsu, W.H.: Drone-based object counting by spatially regularized regional proposal network. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 4165–4173 (2017)

41. Boominathan, L., Kruthiventi, S.S.S., Babu, R.V.: Crowdnet: A deep convolutional network for dense crowd counting. In: Proceedings of the 24th ACM International Conference on Multimedia (2016)

42. Sam, D., Surya, S., Babu, R.: Switching convolutional neural network for crowd counting. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4031–4039. IEEE Computer Society, Los Alamitos (2017)

43. Zeng, L., Xu, X., Cai, B., Qiu, S., Zhang, T.: Multi-scale convolutional neural networks for crowd counting. 2017 IEEE International Conference on Image Processing (ICIP) pp. 465–469 (2017)

44. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015). http://arxiv.org/abs/1409.1556

45. Liu, W., Salzmann, M., Fua, P.V.: Context-aware crowd counting. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5094–5103 (2019)

46. Aich, S., Stavness, I.: Global sum pooling: a generalization trick for object counting with small datasets of large images. arXiv preprint arXiv:1805.11123 (2018)

47. Jiang, X., Xiao, Z., Zhang, B., Zhen, X., Cao, X., Doermann, D.S., Shao, L.: Crowd counting and density estimation by trellis encoder-decoder networks. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6126–6135 (2019)

48. Chen, Z., Cheng, J., Yuan, Y., Liao, D., Li, Y., Lv, J.: Deep density-aware count regressor. In: ECAI (2020)

49. Song, Q., Wang, C., Wang, Y., Tai, Y., Wang, C., Li, J., Wu, J., Ma, J.: To choose or to fuse? Scale selection for crowd counting. In: AAAI (2021)

50. Wang, Q., Breckon, T.: Crowd counting via segmentation guided attention networks and curriculum loss. In: IEEE Transactions on Intelligent Transportation Systems (2022)

51. Guerrero-Gómez-Olmedo, R., Torre-Jiménez, B., López-Sastre, R.J., Maldonado-Bascón, S., Oñoro-Rubio, D.: Extremely overlapping vehicle counting. In: IbPRIA (2015)

Muhammad Asif Khan received the B.Sc. degree in Telecommunication Engineering from University of Engineering and Technology Peshawar, Pakistan (2009) and M.Sc. in Telecommunication Engineering from University of Engineering and Technology Taxila, Pakistan (2013). He received the Ph.D. degree in Electrical Engineering from Qatar University in 2020. He was a Researcher Assistant at Qatar University (2014–2015) and at Qatar Mobility Innovation Center (2016–2017). He is currently working as a postdoctoral researcher at Qatar University. His current research interests include wireless communication, mobile edge computing, and distributed machine learning. He is a Senior Member of IEEE and a Member of IET. For more detailed information, please visit his homepage: www.asifk.me

Hamid Menouar (SM'16) is a Senior Research and Development Expert at Qatar Mobility Innovations Center (QMIC), Qatar, where he leads the development of innovations in the area of smart cities and smart living. Before moving to Qatar in 2010, Hamid worked for 8 years at Hitachi research lab in Sophia Antipolice, France, where he contributed and led research and development activities related to connected vehicles. During his experience with Hitachi, he has heavily contributed to the development of connected vehicles communication standards in Europe through an active participation at ETSI TC ITS and Car to Car Communication Consortium. Hamid received the Engineering degree (M.S.) in computer science from the University of Science and Technology Houari Boumediene, Algiers, Algeria, in 2003, the DEA (M.S.) degree in systems and information technologies from the University of Technology of Compiegne, France, 2004, and the Ph.D. degree in computer science from Telecom ParisTech, Paris, France, in 2008.

Ridha Hamila received the MSc, LicTech with distinction, and PhD degrees from Tampere University of Technology (TUT), Tampere, Finland, in 1996, 1999, and 2002, respectively. Dr. Hamila is currently a Full Professor at the Department of Electrical Engineering, Qatar University, Qatar. From 1994 to 2002 he held various research and teaching positions at TUT within the Department of Information Technology, Finland. From 2002 to 2003 he was a System Specialist at Nokia research Center and Nokia Networks, Helsinki. From 2004 to 2009 he was with Emirates Telecommunications Corporation, UAE. Also, from 2004 to 2013 he was adjunct Professor at the Department of Communications Engineering, TUT. His current research interests include mobile and broadband wireless communication systems, Mobile Edge Computing, Internet of Everything, and Machine Learning. In these areas, he has published over 200 journal and conference papers most of them in the peered reviewed IEEE publications, filed seven US patents, and wrote numerous confidential industrial research reports. Dr. Hamila has been involved in several past and current industrial projects, Ooreedo, Qatar National Research Fund, Finnish Academy projects, EU research and education programs. He supervised a large number of under/graduate students and postdoctoral fellows. He organized many international workshops and conferences. He is a Senior Member of IEEE.