



MACC Net: Multi-task attention crowd counting network

Sahar Aldhaheeri¹ · Reem Alotaibi¹ · Bandar Alzahrani¹ · Anas Hadi¹ · Arif Mahmood² · Areej Alhothali¹ · Ahmed Barnawi¹

Accepted: 1 July 2022 / Published online: 8 August 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Crowd counting and Crowd density map estimation face several challenges, including occlusions, non-uniform density, and intra-scene scale and perspective variations. Significant progress has been made in the development of most crowd counting approaches in recent years, especially with the emergence of deep learning and massive crowd datasets. The purpose of this work is to address the problem of crowd density estimation in both sparse and crowded situations. In this paper, we propose a multi-task attention based crowd counting network (MACC Net), which consists of three contributions: 1) density level classification, which offers the global contextual information for the density estimation network; 2) density map estimation; and 3) segmentation guided attention to filter out the background noise from the foreground features. The proposed MACC Net is evaluated on four popular datasets including ShanghaiTech, UCF-CC-50, UCF-QRNF, and a recently launched dataset HaCrowd. The MACC Net achieves the state of the art in estimation when applied to HaCrowd and UCF-CC-50, while on the others, it obtains competitive results.

Keywords Crowd counting · Crowd classification · Crowd analysis · Attention model · Density map estimation

1 Introduction

Crowd analysis has recently received much interest because of its wide applications, such as video surveillance, public safety, and traffic control. However, accurate crowd counting has been a difficult subject in computer vision due to issues such as occlusions, perspective distortions, size changes, and different crowd distributions. Some early approaches handle the challenge of crowd counting by recognizing each individual pedestrian in a group [1, 2], while others depend on crafted multi-source features [3]. For situations involving high levels of occlusion and heterogeneous crowd distribution, these approaches may perform poorly. To address this, various novel approaches based on convolutional neural networks (CNNs) have recently been proposed for accurate crowd density map estimation and accurate crowd counting. These techniques are primarily intended to address two main challenges: substantial head size fluctuations induced by camera

viewpoint and heterogeneous crowd distributions with high background noise levels. In real-world applications, population density varies drastically between areas and time periods. Even within the same image, the population density may be substantially larger in some places than in others (see Fig. 1). Moreover, predicting density maps for dense crowds is, in general, more challenging than for sparse crowds, resulting in more training loss of the former. As a result, sparse crowd samples are frequently overlooked during training. Sparse crowd counting, on the other hand, may be critical for some applications.

Since the concept of the density map was initially presented in [4], crowd counting has been heavily dominated by density estimation-based approaches. Large-scale datasets, which are widely available [5, 6], together with deep convolutional neural networks have been utilized in density map estimation [7]. To obtain high accuracy, various CNN-based approaches use a multi-column or multi-resolution network to address the issue of scale changes [8]. The size and multi-column structure of hand-crafted filters limit the solutions, even though they increase scale variations. Each column in MCNN [9] is dedicated to a certain level of crowded scene, though Li et al. [10] revealed that each column in a branch structure learns almost identical properties. Hossain et al. 2019 [11] attempted to use an attention mechanism to direct

✉ Sahar Aldhaheeri
saldhaheeri0002@stu.kau.edu.sa



Fig. 1 Examples of challenging crowd counting scenarios: scale variations, occlusion and perspective distortions

the network to automatically focus on particular global and local scales suited for the image. However, because this system only employs the attention model to describe three different scale levels and still relies on a multi-column layout, it struggles to cope with congested crowd scenarios.

To overcome the aforementioned challenges, we present the multi-task attention based crowd counting network (MACC Net). The focus of the architecture is density estimation. Both the classification and segmentation are added to help the model better predict the density map. The practical significance of our work is to extract global and perspective information in crowd scenes. In addition, to filter out background noise from the foreground features, an attention layer guided by the segmentation map is used, which in turn helps in accurately predicting the density map. Meanwhile, the classification phase is used to offer global contextual information to the density estimate network. Moreover, it learns to encode global contextual information by categorizing the density of a particular image patch into five predefined classes inspired by the method of Gao et al. [12]. In the meantime, the segmentation map basically discriminates the regions of background and foreground from the feature maps. Thus, by predicting the segmentation map, the attention layer learns to discriminate background regions from the foreground regions [13]. The following are the major contributions of the current work:

1. A novel multi-task attention based crowd counting framework is proposed that fuses information from multiple sources to make it more resilient to scale variations and background noise.
2. Density based classification task is proposed to capture global contextual information for density map estimation.
3. Segmentation guided attention is proposed to filter background noise from the scene foreground.
4. Wide range of experiments on four state of the art benchmark datasets reveal the strength of the proposed framework.

The rest of this paper is arranged as follows. Section 2 covers related work on crowd counting. We present our suggested framework for multi-tasks with information fusion in Section 3. The experiments and results on four

benchmark datasets are presented in Section 4. Finally, our work is concluded in Section 6.

2 Related work

In this part, we present and discuss some of the most common CNN-based crowd counting and density estimation techniques. In addition, because the proposed MACC Net makes use of classification and attention-based methods, a few related papers are also briefly discussed.

2.1 Crowd counting

The majority of early conventional studies rely on detection-based approaches that use a body or part-based detector to find and count persons in a crowd picture. However, the efficacy of these approaches is limited by occlusions in extremely density settings.

1. **Traditional methods:** Regression-based approaches are used to learn a direct mapping from the extracted feature to the number of objects in order to solve the problem. Idrees et al. [3] suggested a method that extracts features by Fourier analysis and SIFT interest point-based counting in local patches, both of which are related methods, whereas Lempitsky et al. [4] suggested a technique that learns linear mapping between features and their item density maps in the local region due to the neglected saliency that generates erroneous findings in local regions. Furthermore, because learning an ideal linear mapping is challenging, Pham et al. [14] employed random forest regression to learn a non-linear mapping rather than a linear one.
2. **CNN-based methods:** Density map approaches have recently outperformed direct regression in terms of performance as a result of the crowd's spatial distribution. Density map methods employ a density map as an intermediate representation, which is subsequently summed to determine the count for a given region. There are two phases in the general design of density map estimating methods: 1) dot map annotations are used to create density maps; 2) deep learning methods are used

to predict density maps from photos. Convoluting the dot map with a fixed or adaptive Gaussian kernel is the most common way to build a density map. Hand-crafted density maps, on the other hand, may not be optimal in terms of end-to-end learning. For instance, Wan and Chan [15] proposes customized density maps for various networks and datasets. Meanwhile, Ma et al. [16] presents the use of Bayesian loss for calculating the difference between the dot map and the expected density map. To tackle scale variations, enhance findings, conduct domain adaptation, or leverage context information, several deep architectures for estimating density maps have been developed. The use of crowd counting methods requires domain adaptation. Zhang et al. [17] proposes a fine-tuning model with a resampled dataset to adapt new scenes. A GAN-based technique for converting the synthetic dataset to actual datasets is proposed by Wang et al. [18] as well as a new synthetic crowd dataset. Moreover, Kang et al. [19] provides an adaptive convolution method that includes camera parameters as side information.

To decrease estimate error, Walach et al. [20] used layered boosting and selective sampling approaches. Instead of patch-based training, Shang et al. [22] suggested a CNN-based estimate approach that accepts the entire picture as input and directly delivers the final crowd count. To address the issue of scale variation for creating density maps, Boominathan et al. [2] reported the first approach entirely employing convolutional networks and dual-column architecture.

2.2 Attention-based approaches

Attention-based methods take advantage of the visual attention mechanism to direct the attention of the counting network to valuable information in order to increase counting accuracy [21]. Zhang et al. [22] presented the relational attention network (RANet), which captures the interrelation

information of pixels via both local and global self-attention methods, resulting in more informative feature representations. Moreover, Liu et al. [23] presented the attention-injective deformable network (ADCrowdNet), which uses an attention map generator to give areas and congestion degrees for the estimator density map, whereas recurrent attentive zooming network (RAZN), proposed by Liu et al. [23], iteratively identifies locations with significant uncertainty and re-evaluates them in high-resolution space. Other concerns were addressed using visual attention methods, such as background noise in crowded cluster circumstances and different density levels due to size changes [24].

2.3 Crowd classification

The term “crowd density level” is defined as the degree of crowd congestion present in a crowded situation. A continuous (0.0-1.0) or discrete (0-N) value is often used to describe elements of a crowded environment. Wu et al. presented texture analysis characteristics to construct a continuous density level estimation [25]. Meanwhile, for discrete density level classification, Fu et al. produced a deep convolutional neural network [26]. The amount of uncertainty associated with a particular density level estimate is the most significant challenge with this endeavor. There is no consistent system for assigning density level labels and the precise meanings that accompany them across datasets. The most responsive technique is that in which discrete density level labels are directly inferred from authentic crowd count estimate data, culminating in a histogram-style distribution with subjectivity and mishandling minimized to a bare minimum (Fig. 2).

3 Proposed method

Based on the success of cascaded convolutional networks for related multiple tasks [6, 27], we propose two related

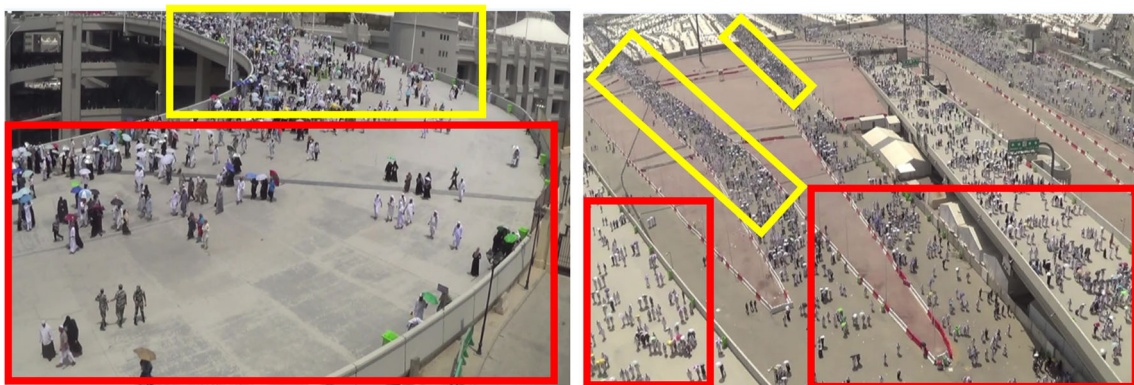


Fig. 2 Examples of scenarios where both sparse and crowded situations occur at the same time. The red boxes represent the sparse part while the yellow boxes refer to the crowded parts

sub-tasks in a cascaded method, as illustrated in Fig. 3. In this section, we describe the flowchart of entire networks and explain detailed information about MACC Net.

3.1 Problem description

Since the inception of density map regression concept, it has dominated crowd counting approaches. The ground-truth density map C_{gt} is obtained by using Gaussian kernels to convey the head count information across surrounding regions for a certain scene. The ground-truth density map in sparse regions is entirely reliant on a few heads, resulting in regular Gaussian blobs. While multiple congested heads may spread to the same surrounding pixel in dense areas, resulting in high ground-truth densities with significantly distinct density patterns than in sparse regions. Because of these variations in density patterns, effectively predicting density maps for both crowded and sparse places is challenging (See Fig. 2). To tackle the problem of pattern shift caused by significant density variations and optimize the predictions for congested regions, we propose Multi-task Attention based Crowd Counting Network (MACC-Net). Before attempting to count the people, the model first partitioning the image into segments of various crowd levels using ROI density estimation. Scale module selects the congested regions (based on ROI density estimation) and zooms the dense areas. As a result, in the enlarged version, the space between surrounding heads is increased, resulting in regular individual Gaussian blobs of target density map M^{den} . In addition, an attention layer led by

the segmentation map is employed to filter out background noise from foreground features, which contributes in properly predicting the density map.

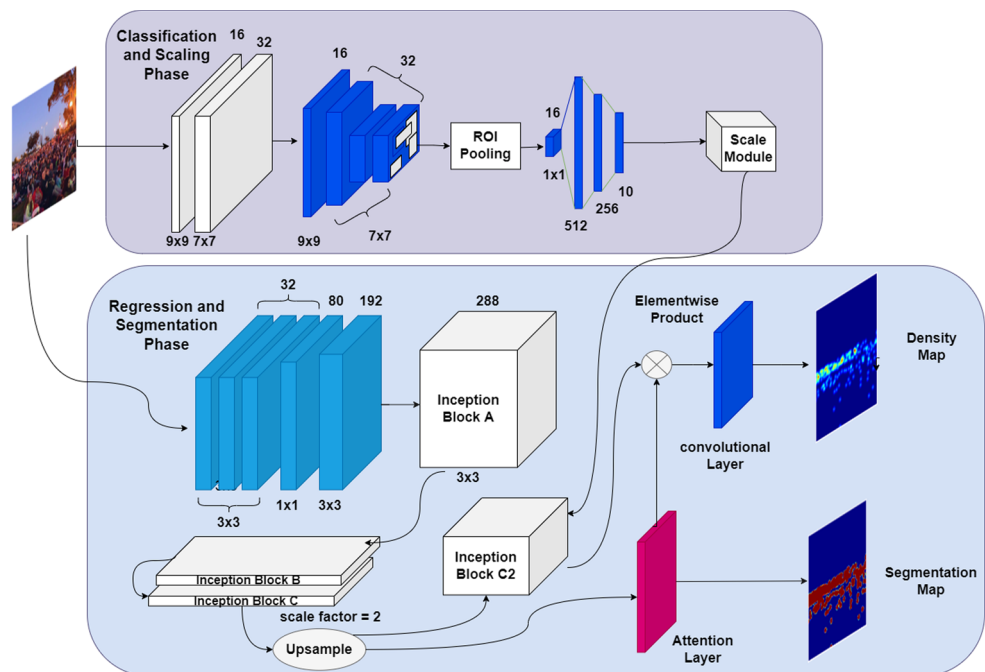
3.2 Proposed MACC-Net framework

The proposed model architecture consists of three separate networks interlinked with each other: the classification model, the segmentation model, and the regression model.

3.2.1 Density level classification

The crowd density in each image is computed using the density map and then divided into 10 levels. During training, the classification net learns to predict the density of the crowd in the image by encoding global contextual information. Density estimation is a pixel-wise regression task that relies on local features and fails to encode global contextual information. As global contextual information is also important, the classification phase is used to provide that contextual information to the density estimation network. The classification phase learns to encode the global contextual information by quantizing the density of a certain image patch into 10 predefined categories. It does so by dividing the input image into a fixed number of patches called regions of interest (ROIs), which are then passed through an ROI pooling layer [28] that converts all the patches into a fixed-length feature map. The fully connected layers are added at the end to predict the density of the crowd in those patches. In order to encode global

Fig. 3 Overall pipeline of the proposed MACC architecture with three tasks: 1) Density level classification to encode the global contextual information and learn to scale and zoom in the dense areas for improved counting accuracy, 2) Density Map Estimation to generate the density map, and 3) Segmentation guided attention which filter out background noise from the foreground features. This mitigate the problem of density pattern shift produced by density differences between sparse and dense regions



contextual information, the image patches are made large enough that only a few (10-15) can cover the whole image. The backbone of the classification network is a custom two-layered (FCN) model that works as a feature extractor. After that, the resultant local features are further passed through four convolutional layers. After the fourth convolutional layer, the path splits into two separate networks: one is the density map and the other is ROI pooling.

As the feature maps produced by the convolution layer immediately before the ROI pooling layer encode global contextual information, those features are directly fed to scale module. After predict the density level and distribution over the image, Inspiring by Xu et al., [29], scale module select the congest regions (based on the ROI density estimation) and zoom in the dense areas for improved counting accuracy. Consequently, the distance between surrounding heads is expanded in the zoomed form, resulting in regular individual Gaussian blobs of target density map M^{den} , mitigating the density pattern shift. Hence, it changes the distance between blobs while preserving the same peaks to scale the ground-truth density map. The module’s objective is to put dense regions of various scales closer at acceptable proximity levels (See Fig. 4). We define a closeness level S for a given area R using ground-truth as follows:

$$S = \frac{\sum_{i=1}^{P_R} d_i}{P_R} \tag{1}$$

where d_i is the distance between the $i - th$ person in R and their nearest neighbor, and P_R is the total number of individuals in the area R . The feature maps produced by this module are directly fed to density map estimation network and are concatenated just before the up-sampling layer. The objective is to train this phase while minimizing standard cross-entropy loss.

3.2.2 Density map estimation

The input images are passed through a modified version of the Inceptionv3 model [30]. As this is a pixel-wise

regression task, we first removed the classification layer at the end of the Inceptionv3 model and only preserved the convolutional layers. The spatial resolution of the output density map is very important, and to preserve it, we decided to remove the first two max-pooling layers.

The output of the final inception module is about 25 times smaller than the input image. The original inception network accepts an image of size 299 x 299 as input, and this means that after the final convolutional layer, the output image will be of size 8 x 8. This is very small, and the density estimation of the original image cannot be directly encoded. To address this, an up-sample layer is added just before the last inception module. As a result, the output image now has a size of 128 x 128 (1/4).

As highlighted above, a number of modifications have been made to the original Inceptionv3 model, but these modifications do not directly change the number of parameters and only increase the number of operations. This also makes it a fully convolutional network (FCN) [31] and, thus, it can process images of any size and generate their corresponding density map.

After the inception module, we have two items, the attention layer and the intermediate high-level features from the classification network. The purpose of the classification network and how it helps the density estimation have already been highlighted in the classification phase (Section 3.2.1). The main item here is the attention layer.

The approach involves using the segmentation map as an attention map for density map estimation. The attention layer, guided by the segmentation map, is used to filter out the background noise from the foreground features, which in turn helps in accurately predicting the density map. The attention layer is trained by learning to directly predict the segmentation map. The segmentation map basically discriminates the regions of background and foreground from the feature maps. Thus, by predicting the segmentation map, the attention layer learns to discriminate background regions from the foreground regions. This idea was proposed by Qian et al. [32].

The attention map generated by the attention layer is then applied to the output from the last inception module using an

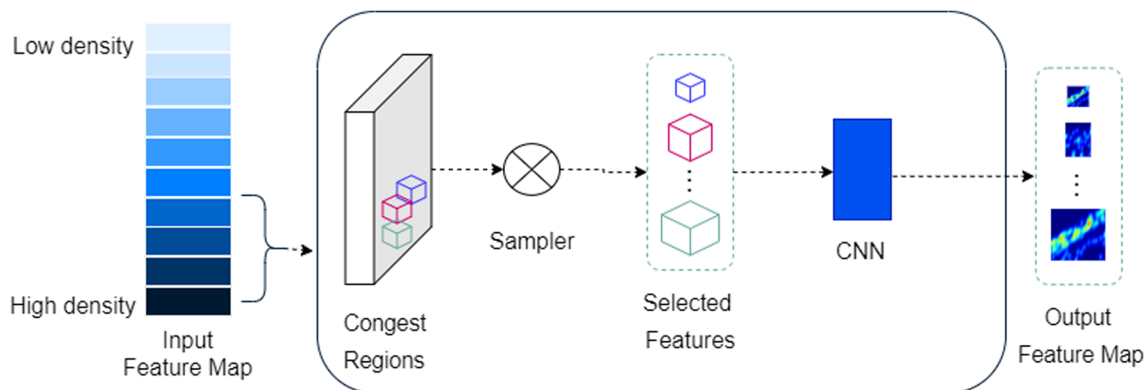


Fig. 4 Detailed architecture of Scale Module

element-wise product. As the output from the Inceptionv3 module might have multiple channels and the attention map only has one channel, the element-wise product is taken with all the channels of the feature map.

The resultant features from the above are further passed through a convolutional layer to predict the final density map. The purpose is to minimize the value of the loss function, which is the standard mean squared error (MSE).

3.2.3 Segmentation guided attention

The main focus of the architecture is density estimation. Both classification and segmentation were added to help the model better predict the density map. The principle for adding the attention approach is to improve counting accuracy, whereby the attention approach uses the visual attention mechanism to guide the counting network's attention to valuable information. Where the crowd density map is typically noisy, crowd segmentation is used as a support job to assist the front-end CNN generate more discriminative representations and, hence, purify the output prediction. The main function of the attention layer is to filter out background noise from the foreground features, guided by the segmentation map, which in turn helps in accurately predicting the density map. Meanwhile, the segmentation map basically discriminates the regions of background and foreground from the feature maps. Thus, by predicting the segmentation map, the attention layer learns to discriminate background regions from the foreground regions. The dotted annotations of the given counting dataset were used to infer ground-truth labels for crowd segmentation by simple binarization. The binary picture is initially created based on the head locations, where the value of key points are 1 and those of other pixels are 0. Although this is a simple strategy, the experiment indicates that it can significantly improve density estimation.

The output from the up-sample layer, just before the final Inceptionv3 block, is passed through an attention layer to directly predict the segmentation map. As discussed in Section 3.2.2, the segmentation map basically acts as an attention map for density estimation. Hence it is recommended that a direct output of the attention layer should be prediction of the segmentation map.

The attention layer is basically a convolution layer. To restrict the output values in the range 0-1, a sigmoid layer is applied to the output of the attention layer. The resultant is the predicted segmentation map.

3.3 Loss functions

The entire model is trained in order to optimize the following unified loss function as described below:

$$L(\theta) = L^{\text{den}}(\theta) + \lambda_1 L^{\text{seg}}(\theta) + \lambda_2 L_\theta \quad (2)$$

where λ_1 and λ_2 are the hyperparameters and the L^{den} , $\lambda_1 L^{\text{seg}}$, and L_θ denote the loss function of the density map, segmentation map, and density level classification, respectively.

3.3.1 Density level classification

In the classification task the objective is to accurately predict the density level of a given patch. The loss used for this task is the weighted cross-entropy loss, which is defined as:

$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top, \quad l_n = - \sum_{c=1}^C w_c \log \frac{\exp(x_{n,c})}{\sum_{i=1}^C \exp(x_{n,i})} y_{n,c} \quad (3)$$

where x is the input, y is the target, w is the weight, C is the number of classes, and N is the mini-batch.

3.3.2 Density map

Density map is a pixel-wise regression task. Hence, we can use a ℓ_2 loss for this phase. To help the model better learn the task, we have decided to use curriculum learning. It builds on the idea of learning the simple things first and then gradually moving toward the more difficult tasks. The images has a diverse category of crowd density. Some images have lower density and some have very high density. Learning to accurately predict the density map for a very dense crowd is a difficult task. Hence, in curriculum learning, the model is first trained to predict the density map for lower density crowd images and higher density images are then gradually included in the training set. As such, one can divide the dataset into different sub-sets with increasing levels of crowd density and, during training, each sub-set can then be gradually added to the training dataset [33]. Instead of dividing the dataset into sub-sets, we are modifying the loss function to incorporate curriculum learning. The loss is known as curriculum loss [4]. As this is a pixel-wise regression task, the curriculum loss is applied to each individual pixel. The pixels of higher values in the density map correspond to the region of the image with dense crowds. During training, a dynamic threshold is used that decides whether a certain pixel is difficult (higher density). The pixels that have a value higher than the dynamic threshold are assigned less weight (less than 1), and the pixels that have a value less than the threshold are assigned more weight (equal to 1). This way, the model mostly focuses on pixels that have pixel values less than the threshold (lower density value). At the start, the threshold is set to the most basic value, and it then gradually increases during training based on the epoch number or the learning curve.

The threshold can be calculated using:

$$T(e) = ke + b \quad (4)$$

Table 1 Main information of the crowd-counting datasets

Dataset	Year	Dens	Res	N	C	Min	Max	Avg	Ref
ShanghaiTech A	2016	Low-D	868x589 (*)	482	~ 240 k	33	3139	501	Zhang et al [9]
ShanghaiTech B	2016	Low-D	1024x768	716	~ 90 k	9	578	123	Zhang et al [9]
UCF-QNRF	2018	Mid-D	2902x2013 (*)	1535	~ 1250 k	49	12865	815	Idrees et al [5]
UCF-CC-50	2013	Ultra-High-D	2888x2101 (*)	50	~ 64	94	4543	1279	Idrees et al [3]
HaCrowd	2021	Mid-D	1286x2178	219	~ 13 k	15	3056	620	HaCrowd [34]

Datasets with different sizes are labeled as (*). Dens, Res, N, C, Min, Max, and Avg denote density level, average resolution, sample number, crowd annotation, minimum, maximum, and average number, respectively

where b is the initial threshold, k is the rate at which the threshold value is to be increased, and e is the training epoch index.

Thus, the above equation is applied to every pixel of the density map in order to generate a weight for each pixel in the density map. The resultant is a weight matrix of the same size as the density map. This weight matrix is then used in the L2 loss function to calculate the overall loss of the predicted density map. The weight matrix is denoted by W and calculated using:

$$W = \frac{T(e)}{\max \{M^{den} - T(e), 0\} + T(e)} \tag{5}$$

where M^{den} is the target density map. This is combined with the L2 loss function to obtain the final loss function for the density map estimation.

$$\mathcal{L}^{den}(\Theta) = \frac{1}{2N} \sum_{i=1}^N \|W(e) \odot (\hat{M}_i^{den} - M_i^{den})\|_F^2 \tag{6}$$

where $W(e)$ is also a function with respect to the training epoch index e , and \odot denotes element-wise multiplication.

3.3.3 Segmentation guided attention map

For the segmentation, we can directly use the cross-entropy loss.

$$\mathcal{L}^{seg}(\Theta) = -\frac{1}{N} \sum_{i=1}^N \|M_i^{seg} \odot \log(\hat{M}_i^{seg}) + (1 - M_i^{seg}) \odot \log(1 - \hat{M}_i^{seg})\|_1 \tag{7}$$

Table 2 The range of crowd counting of each density level on the five datasets

Dataset	None	Low	Medium	High	Very High	Train set	Test set
ShanghaiTech A	0	1 – 9	10 – 21	22 – 47	≥ 48	3249	1089
ShanghaiTech B	0	1 – 2	3 – 4	5 – 10	≥ 11	4833	1611
UCF-QNRF	0	1 – 10	11 – 30	31 – 78	≥ 79	9279	3006
UCF-CC-50	0	1 – 20	21 – 57	58 – 121	≥ 122	342	108
HaCrowd	0	1 – 14	15 – 37	38 – 76	≥ 77	1485	486

where $\| \cdot \|_1$ represents the element-wise matrix norm, and \odot denotes element-wise multiplication of two matrices with the same size. The \mathcal{L}^{seg} and the \mathcal{L}^{den} components are combined during network training.

4 Experiments

In this section, the evaluation metrics and experimental details are first described. The methods are assessed from two perspectives: counting performance and density map quality. To be more explicit, each model or method includes the mean absolute error (MAE) and mean squared error (MSE), which are defined as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^n \|Y_i - \hat{Y}_i\| \tag{8}$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \tag{9}$$

where N denotes the number of images in the testing set, while y_i represents the ground truth of the individual’s number and \hat{y}_i is the estimated count value for the i th testing image.

4.1 Datasets

We provide the results of the proposed model on four publicly available crowd counting datasets, ShanghaiTech [9], UCF-QNRF [5], UCF-CC-50 [3], and HaCrowd [34],

whereby ShanghaiTech is divided into two portions to give five datasets for analysis. Table 1 shows the summary statistics for the different datasets. Moreover, the density level of the five datasets is shown in Table 2.

4.1.1 ShanghaiTech

ShanghaiTech is one of the most comprehensive large-scale crowd counting datasets in recent years, with 1198 photos and 330,165 annotations. The dataset was separated into two portions based on various density distributions: Part A (ShTechA) and Part B (ShTechB). ShTechA includes pictures randomly selected from the Internet, while ShTechB contains images taken on a busy street in a Shanghai metropolitan area. ShTechA has a significantly higher density than ShTechB. This results in a demanding dataset representing a variety of scene types and densities. However, the quantity of photos in various density sets is unequal, resulting in low-density sets for the training and test sets. Nonetheless, the dataset's size shifts and viewpoint distortion bring new difficulties and opportunities for the construction of numerous CNN-based networks.

4.1.2 UCF-QNRF

UCF-QNRF is the most recent and largest crowd dataset available. It is composed of 1535 dense crowd photos collected from Flickr, Google, and Hajj film. The dataset has a larger number of scenes with a greater range of views, lighting fluctuations, and densities, with counts ranging from 49 to 12,865, making it more complex and realistic. Furthermore, the image quality is extremely high, resulting in a variety of head sizes.

4.1.3 UCF-CC-50

UCF-CC-50 contains 50 low-resolution black-and-white photographs of incredibly dense crowd scenes. The number of labeled individuals per image varies from 94 to 4543, with an average of 1280. It has a variety of densities as well as perspective aberrations. This dataset is submitted to

a 5-fold cross-validation technique because it only contains 50 images. Hence, even the most powerful contemporary CNN-based approaches are far from ideal for the its outcomes due to the tiny amount of data.

4.1.4 HaCrowd

The HaCrowd dataset was gathered from Mecca's Hajj scenarios. The data in the initial dataset are from a variety of places. High-definition cameras collect certain photos beneath specified roadways and structures. Some of the films were captured by static surveillance cameras. There are 19 photos and 27 videos in the original material. Each video is between 10 and 30 seconds long and depicts typical Hajj sites, such as Mount Arafat and the Jamarat Bridge. The total number of images in the HaCrowd dataset after converting videos into frames is 219. According to the statistical results, the smallest and highest numbers of people in the HaCrowd dataset come from high-definition images, with 15 and 3056 people, respectively. Unlike previous datasets, the HaCrowd dataset comprises images of the Hajj for a given activity, and images from diverse perspectives are taken simultaneously. HaCrowd is a domain-specific dataset with a high density level and a wide range of perspectives.

4.2 Implementation details

The training and evaluation experiments are conducted on NVIDIA GTX 1080Ti GPU using a PyTorch framework [35]. For training, the "Adam" optimizer [36] is used. During the training stage, the L^{den} , $\lambda_1 L^{\text{seg}}$, and $\lambda_2 L^{\text{cls}}$ are set as 10^{-4} . After each round of 25 epochs, the learning rate is lowered by a factor of 0.5. Image patches with a size of 576×768 , randomly cropped, are then used to train the network. The labels also have the same size (Table 3).

4.3 Ablation study

In Table 4, we compare the estimation errors of each phase, and improved performance can be seen after adding

Table 3 The detailed frames of the different datasets

Dataset	Train set					Test set				
	N	L	M	H	VH	N	L	M	H	VH
ShanghaiTech A	368	699	715	672	716	117	319	303	178	172
ShanghaiTech B	2119	409	482	568	656	1233	607	297	343	364
UCF-QNRF	1829	2033	1986	1930	1950	513	635	655	594	609
UCF-CC-50	29	76	79	77	78	6	19	26	19	38
HaCrowd	368	699	715	672	716	117	319	303	178	172

Table 4 Estimation errors for the proposed method on ShanghaiTech Part A

Methods	MAE	MSE
L^{den}	110.2	171.2
$L^{\text{den}} + \lambda_1 L^{\text{seg}}$	93.8	148.2
$L^{\text{den}} + \lambda_2 L^{\text{cls}}$	89.6	134.4
$L^{\text{den}} + \lambda_1 L^{\text{seg}} + \lambda_2 L^{\text{cls}}$	72.4	125.2

the segmentation guided attention and the density level classification phase. To be precise, we calculated the result for density map estimation, and we then added the segmentation guided attention map and the density level classification. Finally, we computed the result for the whole network. The ablation study performed using the ShanghaiTech Part A dataset to gain a better understanding of the relative contributions of the various components of our method. Table 4 illustrates the MAE and MSE of the proposed method.

The first row in Table 4 reports the results of L^{den} , while the second and third rows presents the two estimation errors for $L^{\text{den}} + \lambda_1 L^{\text{seg}}$ and $L^{\text{den}} + \lambda_2 L^{\text{cls}}$ sequentially. As shown, the two estimation errors of the latter rows (MAE:93.8, MSE: 148.2/ MAE:89.6, MSE: 134.4) are less than that of the first (MAE: 110.2, MSE: 171.2). The last row in the table states the results of $L^{\text{den}} + \lambda_1 L^{\text{seg}} + \lambda_2 L^{\text{cls}}$ (MACC Net), which are (MAE: 72.4, MSE: 125.2).

This experimental result verifies the effectiveness of the MACC Net and demonstrates the importance of the attention layer, which is driven by the segmentation map that filters out background noise from foreground features, allowing the density map to be predicted more precisely. Learning to directly forecast the segmentation map is how the attention layer is trained. The background and foreground areas are separated from the feature maps by the

segmentation map. The attention layer learns to distinguish between background and foreground areas by anticipating the segmentation map.

5 Results

Table 5 lists the experimental results of the proposed MACC Net and some state-of-the-art algorithms, where the best result in each column is in bold, while the second best is italicized. The MACC model achieved competitive performance for all four datasets, as shown in result Table 5. The proposed model has excellent counting ability for the UCF-QNRF dataset, where it has the greatest MAE of 140 and the second best MSE of 240.8. For the HaCrowd dataset, our MACC Net shows the best performance, while the Switching-CNN and CSRNet models are close competitors. However, of the six models tested on the UCF-CC-50 datasets, the PCC Net has the most significant result. For the ShTechA and ShTechB, CSRNet has the best results, although the MACC Net and PCCNet models are close behind. Generally MACC Net model demonstrates a strong ability to detect head regions and regress the head count.

Figure 5 presents the visualization and crowd counting results to provide intuitive evidence of how the attention layer enhances density map estimation. The input image is shown in the first row, while the ground truth is shown in the second. The anticipated density and segmentation map, on the other hand, appear in the third and fourth rows, respectively. For a direct comparison, the real and predicted counts are also displayed on the density maps. The prediction errors for the top four cases are fairly minor. Moreover, the model makes accurate predictions in foreground regions, where in the two top images, the trees in the background are correctly recognized and the model discriminate head location and background.

Table 5 MACC Net performance comparison with the state-of-art on five different datasets

Methods	ShTechA		ShTechB		UCF-QNRF		UCF-CC-50		HaCrowd	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN [9]	110.2	173.2	26.4	41.3	277	426	377.6	509.1	151.8	191.2
CSRNet [11]	68.2	115	10.6	16.0	141.2	242.6	392.9	472.1	<u>55.6</u>	78.7
Cascaded-MTL [37]	101.3	152.4	20.0	31.1	252	514	322.8	341.4	79.4	90.3
Switching-CNN [38]	90.4	135.0	21.6	33.4	228	445	318.1	439.2	66.35	<u>70.02</u>
PCC Net [12]	73.5	124.0	11.0	19.0	148.7	247.3	240.0	<u>315.5</u>	118.3	136
D-ConvNet [39]	73.5	<u>112.3</u>	18.7	26.0	—	—	288.4	404.7	80.34	88.11
SARM [40]	64.4	100.2	8.4	13.4	—	—	<u>242.3</u>	320.4	—	—
AU-CNN [41]	70.4	117.5	<u>8.6</u>	<u>13.4</u>	—	—	243.9	320.4	—	—
MACC (Our)	72.4	125.2	11.3	15.09	140.3	<u>240.8</u>	289.3	316.2	54.97	<u>69.5</u>
MACC + SM (Our)	<u>67.7</u>	113	9.8	12.9	<u>140.7</u>	238	275.8	311	55.9	68.9

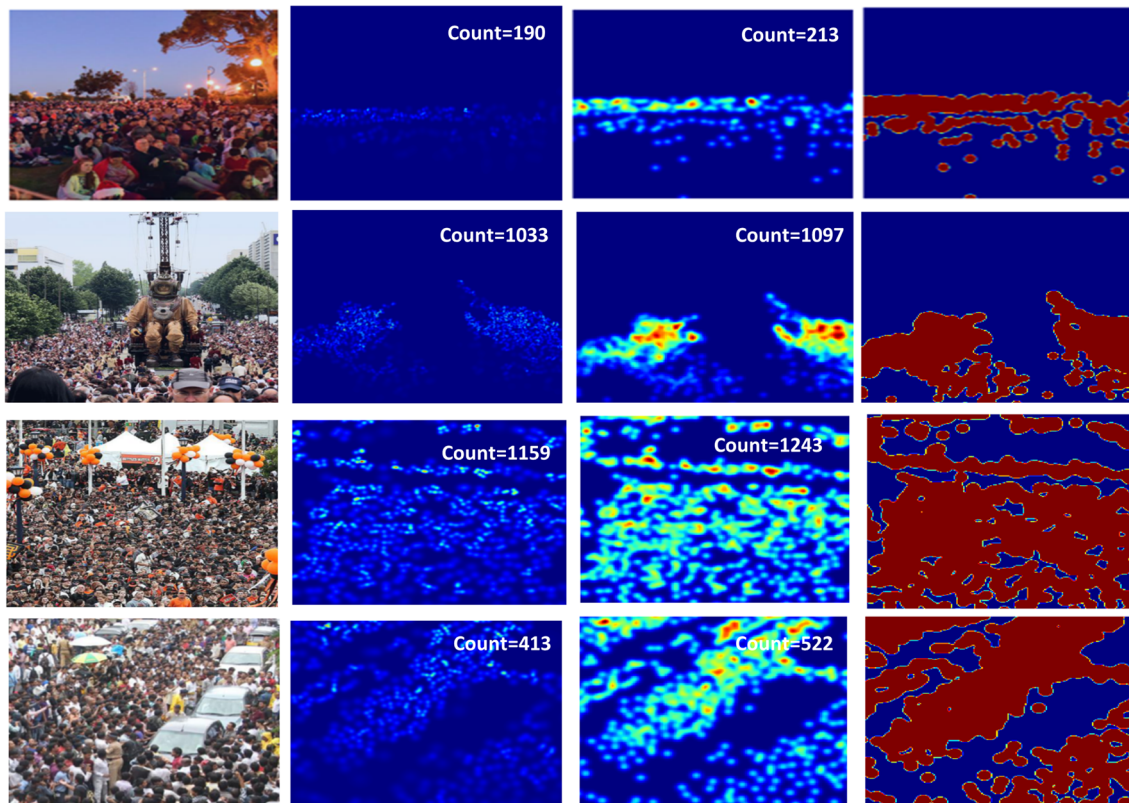


Fig. 5 Visualization of density and segmentation maps for images from ShanghaiTech part A and HaCrowd dataset. Row 1: Input images; Row 2: Groundtruth; Row 3: Predicted density map; Row 4: Predicted segmentation map

6 Conclusions

In this paper, we proposed a multi-task attention crowd counting network (MACC Net). We investigated the effectiveness of learning crowd classification and density map estimation simultaneously. We include high-level features into the network beforehand, allowing it to learn globally relevant discriminative features and, hence, account for significant count changes in the dataset. The MACC Net consists of density level classification, density map estimation (DME), and segmentation guided attention. Density level classification uses global characteristics to estimate the coarse density labels of random picture patches. Meanwhile, the attention layer, which is driven by the segmentation map, filters out background noise from foreground features, allowing the density map to be predicted more precisely. The MACC Net reaches the state of the art on HaCrowd and UCF-CC-50 datasets, while it obtains competitive results on the others.

Acknowledgments The Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia has funded this project, under grant no. (FP-090-43).

References

1. Ge W, Collins RT (2009) Marked point processes for crowd counting. In: 2009 IEEE conference on computer vision and pattern recognition, CVPR 2009, pp 2913–2920. <https://doi.org/10.1109/CVPRW.2009.5206621>
2. Li M, Zhang Z, Huang K, Tan T (2008) Estimating the number of people in crowded scenes by MID based foreground segmentation and head-shoulder detection. In: Proceedings - international conference on pattern recognition. <https://doi.org/10.1109/icpr.2008.4761705>
3. Idrees H, Saleemi I, Seibert C, Shah M (2013) Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, pp 2547–2554. <https://doi.org/10.1109/CVPR.2013.329>
4. Lempitsky V, Zisserman A (2010) Learning to count objects in images. In: Advances in neural information processing systems 23: 24th annual conference on neural information processing systems 2010, NIPS 2010, vol 23
5. Idrees H, Tayyab M, Athrey K, Zhang D, Al-Maadeed S, Rajpoot N, Shah M (2018) Composition loss for counting, density map estimation and localization in dense crowds. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), vol 11206 LNCS, pp 544–559. https://doi.org/10.1007/978-3-030-01216-8_33
6. Zhang K, Zhang Z, Li Z, Qiao Y (2016) Joint face detection and alignment using Multitask cascaded convolutional networks.

- IEEE Signal Process Lett 23(10):1499–1503. arXiv:1604.02878, <https://doi.org/10.1109/LSP.2016.2603342>
7. Kravchik M, Shabtai A (2018) Detecting cyber attacks in industrial control systems using convolutional neural networks. In: Proceedings of the ACM Conference on computer and communications security. p 72–83, association for computing machinery. <https://doi.org/10.1145/3264888.3264896>
 8. Oñoro-Rubio D, López-Sastre RJ (2016) Towards perspective-free object counting with deep learning. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), vol 9911 LNCS, p 615–629. Springer. https://doi.org/10.1007/978-3-319-46478-7_38. https://link.springer.com/chapter/10.1007/978-3-319-46478-7_38
 9. Zhang Y, Zhou D, Chen S, Gao S, Ma Y (2016) Single-image crowd counting via multi-column convolutional neural network. Proceedings of the IEEE Computer society conference on computer vision and pattern recognition 2016-Decem:589–597. <https://doi.org/10.1109/CVPR.2016.70>
 10. Li Y, Zhang X, Chen D (2018) CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes. In: Proceedings of the IEEE Computer society conference on computer vision and pattern recognition. p 1091–1100, IEEE Computer Society,???. <https://doi.org/10.1109/CVPR.2018.00120>
 11. Hossain MA, Hosseinzadeh M, Chanda O, Wang Y (2019) Crowd counting using scale-aware attention networks. In: Proceedings - 2019 IEEE Winter Conference on Applications of Computer Vision, WACV 2019, pp 1280–1288. <https://doi.org/10.1109/WACV.2019.00141>
 12. Gao J, Wang Q, Li X (2020) PCC Net: Perspective crowd counting via spatial convolutional network. IEEE Trans Circuits Syst Video Technol 30(10):3486–3498. arXiv:1905.10085. <https://doi.org/10.1109/TCSVT.2019.2919139>
 13. Wang Q, Breckon TP (2022) Crowd counting via segmentation guided attention networks and curriculum loss. IEEE Transactions on Intelligent Transportation Systems, p 1–11. arXiv:1911.07990. <https://doi.org/10.1109/tits.2021.3138896>
 14. Pham VQ, Kozakaya T, Yamaguchi O, Okada R (2015) COUNT forest: Co-voting uncertain number of targets using random forest for crowd density estimation. In: Proceedings of the IEEE international conference on computer vision, vol 2015 Inter, pp 3253–3261. <https://doi.org/10.1109/ICCV.2015.372>
 15. Wan J, Chan A (2019) Adaptive density map generation for crowd counting. In: Proceedings of the IEEE international conference on computer vision, vol 2019-Octob, pp 1130–1139. <https://doi.org/10.1109/ICCV.2019.00122>
 16. Zhang Y, Zhao H, Duan Z, Huang L, Deng J, Zhang Q (2021) Congested crowd counting via adaptive multi-scale context learning. Sensors 21(11):3777. <https://doi.org/10.3390/s21113777>
 17. Zhang C, Li H, Wang X, Yang X (2015) Cross-scene crowd counting via deep convolutional neural networks. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, vol 07-12-June, p 833–841. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2015.7298684>
 18. Wang Q, Gao J, Lin W, Yuan Y (2019) Learning from synthetic data for crowd counting in the wild. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, vol 2019-June, p 8190–8199. IEEE computer society. <https://doi.org/10.1109/CVPR.2019.00839>
 19. Kang D, Dhar D, Chan AB (2020) Incorporating Side Information by Adaptive Convolution. Int J Comput Vis 128(12):2897–2918. <https://doi.org/10.1007/s11263-020-01345-8>
 20. Walach E, Wolf L (2016) Learning to count with CNN boosting. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), vol 9906 LNCS, p 660–676. Springer. https://doi.org/10.1007/978-3-319-46475-6_41
 21. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems, vol 2017-Decem, p 5999–6009. Neural information processing systems foundation???? arXiv:1706.03762v5
 22. Zhang A, Shen J, Xiao Z, Zhu F, Zhen X, Cao X, Shao L (2019) Relational attention network for crowd counting. In: Proceedings of the IEEE International conference on computer vision, vol 2019-Octob, p 6787–6796. <https://doi.org/10.1109/ICCV.2019.00689>
 23. Liu N, Long Y, Zou C, Niu Q, Pan L, Wu H (2019) Adcrowdnet: An attention-injective deformable convolutional network for crowd understanding. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, vol 2019-June, p 3220–3229. IEEE computer society. <https://doi.org/10.1109/CVPR.2019.00334>. arXiv:1811.11968v5
 24. Jiang X, Zhang L, Xu M, Zhang T, Lv P, Zhou B, Yang X, Pang Y (2020) Attention scaling for crowd counting. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, p 4705–4714. IEEE computer society. <https://doi.org/10.1109/CVPR42600.2020.00476>
 25. Wu X, Liang G, Lee KK, Xu Y (2006) Crowd density estimation using texture analysis and learning. In: 2006 IEEE International Conference on Robotics and Biomimetics, ROBIO 2006, p 214–219. <https://doi.org/10.1109/ROBIO.2006.340379>
 26. Fu M, Xu P, Li X, Liu Q, Ye M, Zhu C (2015) Fast crowd density estimation with convolutional neural networks. Eng Appl Artif Intell 43:81–88. <https://doi.org/10.1016/j.engappai.2015.04.006>
 27. Chen JC, Kumar A, Ranjan R, Patel VM, Alavi A, Chellappa R (2016) A cascaded convolutional neural network for age estimation of unconstrained faces. In: 2016 IEEE 8th International conference on biometrics theory, applications and systems, BTAS 2016. Institute of electrical and electronics engineers Inc. <https://doi.org/10.1109/BTAS.2016.7791154>
 28. Girshick R (2015) Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. <https://github.com/rbgirshick/>. Accessed 11 April 2022
 29. Xu C, Liang D, Xu Y, Bai S, Zhan W, Bai X, Tomizuka M (2022) AutoScale: Learning to Scale for Crowd Counting. Int J Comput Vis 130(2):405–434. <https://doi.org/10.1007/s11263-021-01542-z>
 30. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, vol 2016-Decem, p 2818–2826. IEEE computer society. <https://doi.org/10.1109/CVPR.2016.308>
 31. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, vol 07-12-June, p 431–440. IEEE computer society. <https://doi.org/10.1109/CVPR.2015.7298965>
 32. Wang Q, Breckon TP (2022) Crowd counting via segmentation guided attention networks and curriculum loss. IEEE Transactions on Intelligent Transportation Systems, p 1–11. arXiv:1911.07990. <https://doi.org/10.1109/tits.2021.3138896>
 33. Jiang L, Meng D, Zhao Q, Shan S, Hauptmann AG (2015) Self-Paced Curriculum learning proceedings of the AAAI conference on artificial intelligence 29(1)
 34. HaCrowd <https://github.com/KAU-Smart-Crowd/HaCrowd> Accessed 11 Nov. 2022
 35. Pytorch (2019) PyTorch: tensors and dynamic neural networks in Python with strong GPU acceleration. <https://github.com/pytorch/pytorch> Accessed 11 Nov. 2022

36. Kingma DP, Ba JL (2015) Adam: A method for stochastic optimization. In: 3rd International conference on learning representations, ICLR 2015 - conference track proceedings. international conference on learning representations, ICLR., arXiv:1412.6980v9
37. Sindagi VA, Patel VM (2017) 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 2017. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/AVSS.2017.8078491>
38. Sam DB, Surya S, Babu RV (2017) Switching convolutional neural network for crowd counting. In: Proceedings - 30th IEEE conference on computer vision and pattern recognition, CVPR 2017, vol 2017-Janua, p 4031–4039. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/CVPR.2017.429>
39. Shi Z, Zhang L, Liu Y, Cao X, Ye Y, Cheng MM, Zheng G (2018) Crowd Counting with Deep Negative Correlation Learning. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, p 5382–5390. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2018.00564>
40. Liu YB, Jia RS, Liu QM, Zhang XL, Sun HM (2021) Crowd counting method based on the self-attention residual network. *Appl Intell* 51(1):427–440. <https://doi.org/10.1007/s10489-020-01842-w>
41. Wu D, Fan Z, Cui M (2022) Average up-sample network for crowd counting. *Appl Intell* 52(2):1376–1388. <https://doi.org/10.1007/s10489-021-02470-8>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Sahar Aldhaheeri received the M.Sc. degree in Information systems from the college of computer and information technology, King Abdulaziz university, KSA, in 2020. She is currently an Assistant Researcher with the Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. Her research interests focused on artificial intelligence and cyber security.

Reem Alotaibi is an associate professor at the Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. Currently, she is the supervisor of the Information Technology department. Dr. Alotaibi received her PhD in computer science from University of Bristol, Bristol, U.K., in 2017. During 2017–2018 she was a visiting lecturer at the Intelligent Systems Laboratory, University of Bristol. Her research interests include Artificial Intelligence and Machine learning.

Bander Alzahrani received his M.Sc. in Computer Security and Ph.D. in Computer Science from University of Essex, UK, in 2010 and 2015, respectively. He is currently an Associate Professor in the Faculty of Computing and Information Technology, King Abdulaziz University, Saudi Arabia. He has led more than 10 national research projects and co-authored more than 70 research articles in peer reviewed journals and conferences. His current research interests include WSN, ICN, secure content routing, intelligent crowd management.

Anas Hadi received the M.Sc. and Ph.D. degrees in computer science from KAU, in 2013 and 2019, respectively. He is currently an Assistant Professor with the Computer Science Department, Faculty of Computing and Information Technology, King Abdulaziz University. His current research interests include machine learning, optimization, brain-computer interface (BCI), and feature selection.

Arif Mahmood is a Professor and Chairperson of Computer Science Department in Information Technology University and Director Computer Vision Lab. His current research directions in Computer Vision are person pose detection and segmentation, crowd counting and flow detection, background-foreground modeling in complex scenes, object detection, human-object interaction detection and abnormal events detection. He is also actively working in diverse Machine Learning applications including cancer grading and prognostication using histology images, predictive auto-scaling of services hosted on the cloud and the fog infrastructures, and environmental monitoring using remote sensing. He has also worked as a Research Assistant Professor with the School of Mathematics and Statistics, University of the Western Australia (UWA) where he worked on Complex Network Analysis. Before that he was a Research Assistant Professor with the School of Computer Science and Software Engineering, UWA and performed research on face recognition, object classification and action recognition.

Areej Alhothali is an assistant professor in the faculty of computer science and information technology at King Abdul-Aziz University. She earned her master's and Ph.D. degrees in computer science (artificial intelligence) from the University of Waterloo, Canada. Her research interest lies in the areas of machine learning, deep learning, natural language processing, intelligent agent systems, affective computing, and sentiment analysis.

Ahmed Baranawi is currently a professor at the Faculty of Computing and IT in King Abdulaziz University. He is the managing director of the KAU Cloud Computing and Big Data Research Group. He received his Ph.D. from the University of Bradford, UK, in 2005 and his M.Sc. from UMIST, UK, in 2001. His research interests include big data, cloud computing, and advanced mobile robotic applications. He has published more than 100 papers in peer reviewed journals.

Affiliations

Sahar Aldhaheeri¹  · Reem Alotaibi¹ · Bandar Alzahrani¹ · Anas Hadi¹ · Arif Mahmood² · Areej Alhothali¹ · Ahmed Barnawi¹

Reem Alotaibi
ralotibi@kau.edu.sa

Bandar Alzahrani
baalzahrani@kau.edu.sa

Anas Hadi
amohammed2@kau.edu.sa

Arif Mahmood
arif.mahmood@itu.edu.pk

Areej Alhothali
aalhothali@kau.edu.sa

Ahmed Barnawi
ambarnawi@kau.edu.sa

¹ Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia

² Department of Computer Science, Information
Technology University, Lahor, Pakistan