International Journal of Computational Intelligence Systems (2021) 14:168
https://doi.org/10.1007/s44196-021-00016-x

RESEARCH ARTICLE

A Deep-Fusion Network for Crowd Counting in High-Density Crowded Scenes

Sultan Daud Khan[1] · **Yasir Salih**[2] · **Basim Zafar**[2] · **Abdulfattah Noorwali**[3]

Received: 16 February 2021 / Accepted: 26 August 2021 / Published online: 28 September 2021

Abstract
People counting has been investigated extensively as a tool to increase the individual's safety and to avoid crowd hazards at public places. It is a challenging task especially in high-density environment such as Hajj and Umrah, where millions of people gathered in a constrained environment to perform rituals. This is due to large variations of scales of people across different scenes. To solve scale problem, a simple and effective solution is to use an image pyramid. However, heavy computational cost is required to process multiple levels of the pyramid. To overcome this issue, we propose deep-fusion model that efficiently and effectively leverages the hierarchical features exits in various convolutional layers deep neural network. Specifically, we propose a network that combine multiscale features from shallow to deep layers of the network and map the input image to a density map. The summation of peaks in the density map provides the final crowd count. To assess the effectiveness of the proposed deep network, we perform experiments on three different benchmark datasets, namely, UCF_CC_50, ShanghaiTech, and UCF-QNRF. From experiments results, we show that the proposed framework outperforms other state-of-the-art methods by achieving low Mean Absolute Error (MAE) and Mean Square Error (MSE) values.

Keywords Crowd counting · Feature fusion · Convolutional neural network

1 Introduction

Automated crowd analysis is crucial for efficient crowd management. Crowd analysis has numerous application, such as panic detection [64], crowd behavior understanding [44, 57], crowd tracking [2], crowd flow segmentation [1], crowd congestion detection [33], and crowd counting [46, 62]. Among these application, crowd counting problem has received tremendous attention from different researchers. This is due to reason that crowd counting can have potential applications in crowd surveillance and scene understanding. For effective crowd surveillance, it is imperative to predict the actual count and location of individuals in the scene. Crowd counting provides support in managing massive crowd, for example, during Hajj and Umrah, where millions of Muslims (from all over the world) gather in Holy city of Makkah to perform obligatory rituals [18]. It is the priority of the Saudi government to ensure smooth conduct of Hajj and Umrah. Therefore, researchers have proposed different automated methods [32] for efficient crowd management. For efficient crowd management, one of the priorities is to estimate crowd count and know distribution of people in the environment, which is the prime goal of this paper.

The goal of a crowd counting system is to count pedestrians in given images/videos irrespective of the scene and density. However, crowd counting is a difficult job, as it offers many challenges, such as variations in scales, clutter background, perspective distortions, and low-resolution images [56], as shown in Fig. 1. Among these challenges, scale variations are a challenging problem [65] and have not been effectively addressed so far for crowd counting problem. Scale variations refers to the variations in object's size (in our case, heads). These variations are due the perspective distortions caused by the location of the camera relative to the scene.

Several attempts have been made to solve the scale problem. A simple and straightforward solution is to re-size the image to different scales and learn multiple object detectors. Each detector will detect an object that falls in its scale

✉ Sultan Daud Khan
sultandaud@nutech.edu.pk

[1] National University of Technology, Islamabad, Pakistan

[2] Expert Vision Consulting, Makkah, Saudi Arabia

[3] Umm Al-Qura University, Makkah, Saudi Arabia

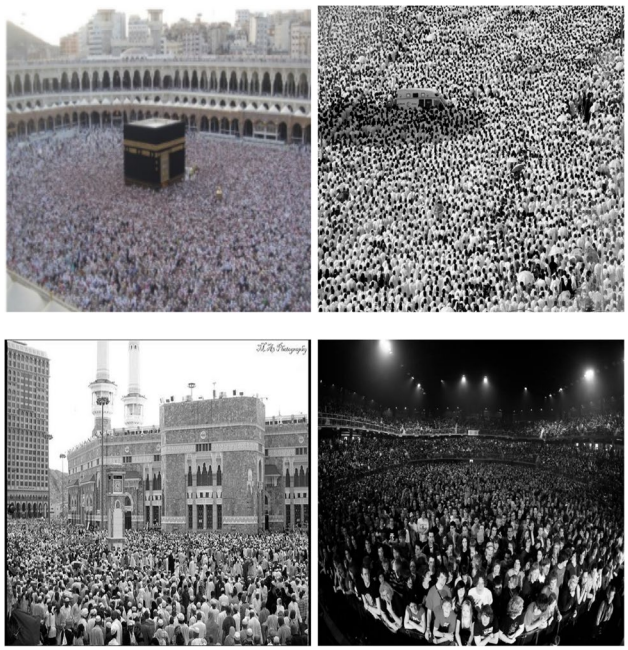


Fig. 1 Sample frames from the datasets

range. Another simple way is to generate hand-crafted feature pyramid to detect different objects of different scales [16]. However, processing all scales of the pyramid leads to computational cost.

Due to the success of Deep Convolutional Neural Networks (DCNNs) in various tasks of computer vision, for example, object detection, classification and segmentation, researchers employ various CNN architectures to learn non-linear function from images to crowd count. However, CNN cannot implicitly handle the scale variations [25]. To make CNN adaptive to scale variations, it must be trained to capture scale variations to a certain extent. For example, a scale-aware network is proposed in [37] that resizes the input image in a way to bring all objects to similar scale and then trained a single-scale detector. Other recent methods [23, 26, 48] utilize the feature maps of the top layers to detect objects of different sizes. Generally, the receptive field of the top-most layers is large and contains little or no details about the small objects. Therefore, it compromises the performance of a detector in detecting small objects, especially people in high-density crowds. Furthermore, these networks require more parameters and have complex architecture to obtain desirable performance.

To handle the scale problem, we propose a framework that fuses feature maps from different layers for crowd counting. Specifically, the proposed framework consist of two parts, i.e., encoder and decoder. In the encoder part, the framework adopts fusion strategy that combines features from multiple layers to capture the details of heads of multiple scales. The decoder part estimates the crowd count and generates density map, where high peaks indicate the presence of human heads in that particular location.

The contribution of the proposed framework can be summarized as follows:

- To capture the information of human heads of multiple scales, we proposed a fusion strategy that combines the information from different layers of the network.
- The framework estimates the crowd density and crowd count simultaneously. The framework predicts density and count both in low- and high-density crowded scenes irrespective of the scene and density.
- The propose method has been tested on different public benchmark datasets. From experiment results, we demonstrate that the effectiveness of proposed framework achieves superior performance compared to other reference methods.

2 Related Works

Crowd counting or density estimation methods can be broadly categorize into two main groups: *holisticapproach* and *localapproach*. In a holistic approach, global features of the image, i.e., textures, edges, and foreground pixels, are extracted from the image and a classifier/regression-based model is trained to learn the mapping between the features and actual crowd size. On the contrary, local approach utilizes the local features of image which are specific to individuals or group of people. We provide details of these approaches in the following subsections.

2.1 Holistic Approaches

Holistic approaches estimate the crowd size by utilizing the global image features. Features utilized by these methods include textures [42], foreground pixels [17], and edge features [34]. The methods proposed in [40, 42] utilize gray level co-occurrence matrix (GLCM) for estimating the crowd density. Minkowski fractal dimension method is proposed in [41] for extracting texture features. Xiaohua et al. [61] achieve classification accuracy of 95% by employing the wavelet descriptors and SVM to classify the crowd density into four classes. However, this method performs well in low-density crowds, however, faces challenges when applied on high-density crowded scenes. Rahmalan et al. [27] proposed a method that achieved a superior performance on afternoon scene, due to less variations in illumination when compared to the morning dataset. This highlights the drawbacks of using texture features when employed in real-time situations, since texture features are not robust to illumination changes.

Other approaches utilize foreground pixels and edges to estimate the crowd size. For example, Regazzoni et al. [47] combined multiple edge detectors, such as vertical edges for detecting the legs and arms of the individuals. Davies et al. [17] found the correlation between the size of crowd and pixels belonging to foreground to establish a principle that the number of people is linearly proportional to foreground pixels and number of edges. Such features are also used in [11–13], where crowd size is estimated by employing a feed-forward neural network. The above-mentioned approaches rely on the static background, where the scenes are relatively captured at a high camera angle.

However, foreground pixels alone cannot provide sufficient information about the number of people in the scene, since the objects in the distance are smaller and will be represented by less number of pixels from the foreground. Therefore, as a solution, Ma et al. [38] propose a crowd estimation framework that incorporates perspective distortions. However, this method cannot handle partial or full occlusions. Similarly, Roqueiro et al. [50] applied the Median Background computing technique to define the foreground pixels. A threshold value is applied on the pixels followed by a morphological operations to smooth the results. A classifier was then trained to categorize the images as either contain zero persons or one or more persons. Similarly, [6–9] adopt holistic approaches to count the number of people in scene and account for occlusion and other non-linearities.

To summarize the discussion, holistic approaches tend to estimate the crowd size by exploiting global features of image. However, due to high variations in crowd dynamic, distribution, and density, crowd size is difficult to estimate. Therefore, as a solution, local approaches are proposed to overcome the limitations of global approaches.

2.2 Local Approaches

These methods use local features that are associated with pedestrian or groups of pedestrians with an image. These approaches can further be sub-categorized into two groups: (i) *detection-based* approaches use head, face to localize the individual in an image, where total number of detections represent the crowd count; (ii) *localization-based* method divides image into overlapping [24, 29, 62] or non-overlapping patches [10, 51, 63], and then, features are computed from each patch and crowd size is predicted by applying regression model.

Detection-based approaches are suitable to the scenes where the crowd is spare, i.e., the people in the scene are well separated and their bodies are fully visible. Therefore, pedestrian detectors/head detectors [16, 19, 21] are employed to get the crowd count. These methods work well in low- density crowds, where pedestrians are not occluded; however, in the real-time environment, pedestrians are always occluded and their bodies are not visible enough that can be detected by pedestrian detection methods. Therefore, as alternative, localization-based methods are proposed which divide the input image into a number of overlapping/non-overlapping sub-regions, where counting is done in each region by employing regression model. For example, localization is performed by employing key points clustering method in [14, 14, 14, 15]. In these methods, SURF features are extracted from an input image. Stationary points are removed by taking the mask of features points with optical flow. The remaining features are clustered into different groups by employing K-means algorithm. The group size is then estimated by employing a regression model. The shortcoming of these approaches is that these methods are restricted to moving objects and could not count the people who are stationary in the scene.

2.3 Convolution Neural Networks (CNN)

Deep Convolutional Neural Networks have achieved remarkable success in various fields of computer vision such as detection, classification, and semantic segmentation. Some researchers made proposed different deep learning frameworks for crowd counting in the recent two years. For crowd counting, Wang et al. [60] proposed first regression base CNN model. Fu et al. [22], on the other hand, proposed a deep convolutional network that classifies the input image into five classes. Shang et al. [53] leverage contextual information at both local and global levels estimate the crowd count by employing end-to-end CNN. Zhang et al. [63] proposed architecture that consists of multiple column, where each column implements a CNN each having different receptive fields to capture scale variations caused by perspective distortions. The network takes an input image of arbitrary size and predicts corresponding density map. Onoro-Rubio et al. [45] proposed a scale-aware crowd density estimation model, Hydra CNN that estimates crowd densities in complex crowded scenes without the need of geometric information of the scene. Sang et al. [52] propose a method, namely, SaCNN that estimates high-quality density maps, where crowd count is obtained by the integrating these density maps. Sindagi et al. [55] proposed end-to-end cascaded CNN that simultaneously estimate the crowd count and density maps. Liu et al. [36] proposed DecideNet that separately generates different density maps. Attention module is used to obtain final crowd count from these two different density maps. Zhang et al. [63] estimated the number of people in a single image using a Convolutional Neural Networks (CNNs) regression model with two configurations. One is a network to estimate head count from a given image, while the other one is to construct the density map of the crowd. The final count was obtained by integrating both output. Kang et al. [31] proposed a crowd segmentation approach by

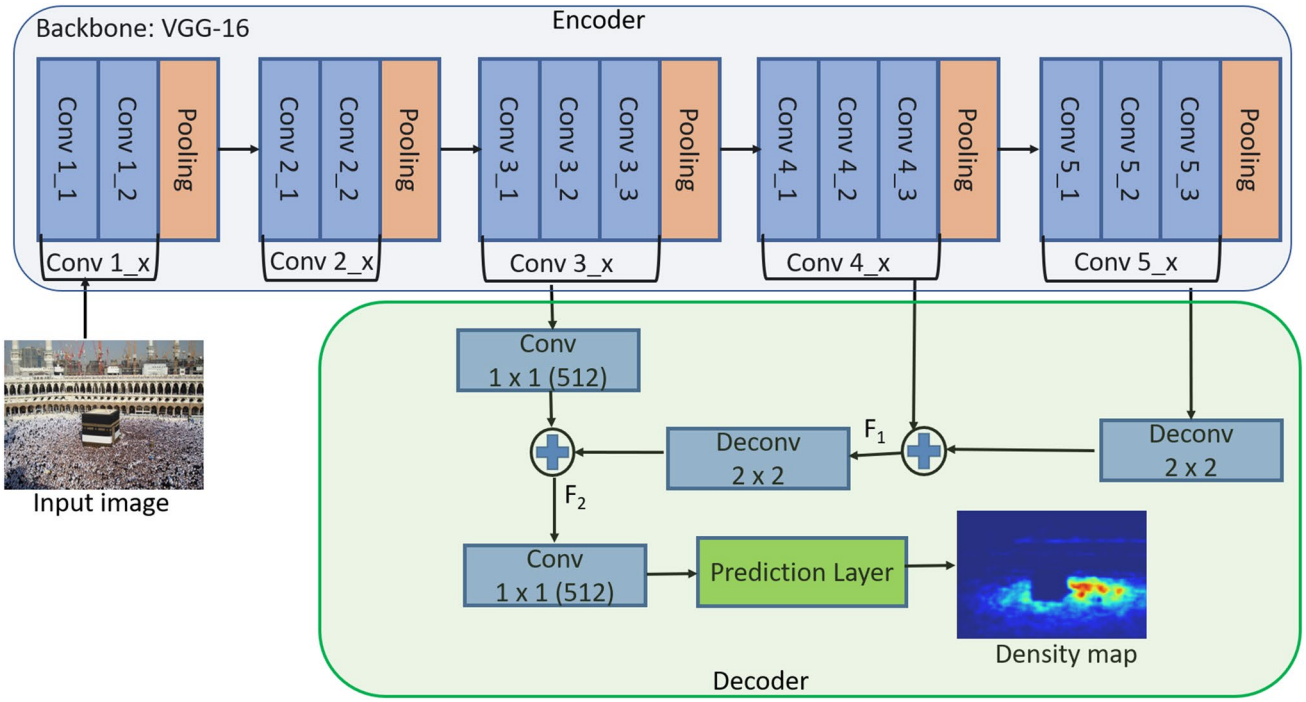


Fig. 2 Architecture of proposed framework for crowd counting and density estimation. The input is the arbitrary size image and output is density map. The summation of density map provides the final crowd count in the given image

constructing a fully convolutional neural network (FCNN) based on both appearance features and motion features. They used one layer of a convolution kernel instead of the fully connected layers in the original CNNs to define the labels at each pixel in the segmentation map. The output is a segmentation map with different probabilities of the crowd on each pixel. Boominathan et al. [4] use fully convolutional network and combine both deep and shallow to estimate the crowd density from crowded images. The shallow network was designed with three convolutional layers to detect the small head blobs arising from people away from the camera. They concatenated the predictions from both networks to predict the crowd density. Crowd count was then obtained by a linear summation of the peaks of the predicted density map.

3 Proposed Methodology

In this section, we provide the details of proposed framework for estimating crowd count in complex scenes. The proposed crowd counting architecture shown in Fig. 2 uses feature maps from different levels that represent features at different scales that are fused together for crowd counting task. Generally, the proposed architecture follows the pipeline of popular crowd density estimation methods [35, 59, 60] that comprise of two networks. The first network is an encoder that takes the input image, and extracts multilevel features from the input image, and second network is a decoder stage that generates density map. The final density map represents the count of people per-pixel in the input image (Fig. 3).

The goal of counting framework is to estimate the distribution of people in input image by optimizing a defined loss function. Similar to other crowd counting framework, our also follows the architecture of VGG-16 [54]. VGG-16 is a popular state-of-the-art CNN and received tremendous success in numerous image classification tasks. The architecture of VGG-16 is divided into five covolutional blocks, where each convolutional block is followed by a max-pooling layer. The receptive field size of all convolutional layers is set to smallest size of 3×3 pixels with stride of 1. The size of max-pooling layer is 2×2 with stride of 2.

The first convolutional block is represented by conv1_x comprising of two convolutional layers with filter size of 3×3 and containing 64 channels each. The first covolutional block is succeeded by a max-pooling layer. The second covolutional block is represented by conv2_*x* and also consists of two convolutional layers with the same filter size (3×3) and the number of channels in each convolutional layer of block conv2_*x* is 128. The second convolutional block is followed by another max-pooling layer of size 2×2 and stride of 2. The third convolutional block conv3_*x* comprises of three convolutional layers, each with filter size of 3×3 and 256 channels. A Max-pooling layer is applied after fourth

Fig. 3 **a** Original image. **b** Points (location of pedestrians) overlaid over the image. **c** Ground-truth density map generated using Eq. 2

convolutional block (conv4_*x*) that consists of set of three convolutional layers of filter size 3 x 3 and consist of 512 channels. The fifth convolutional block (conv5_*x*) is similar to conv4_*x*, followed by the 5^{th} max-pooling layer. The stack of five convolutional blocks is followed by three fully connected layers. The spatial resolution is reduced by 1/2 after passing through each convolutional block; however, the spatial resolution of the feature map is intact inside the block.

Crowd counting methods [59, 60] utilize the features from the last (5th) convolutional layer. However, the last layers of deep neural networks contain rich contextual information, but less details about the small objects due to large receptive fields. These higher order layers can be used to capture the global context of the scene. Furthermore, the resolution of feature maps after subsequent convolutional and pooling layers is reduced that results in poor localization. On the other hand, the shallow layers contains rich information about small objects due to small receptive field. The resolution of feature maps of these layers is large; however, the feature maps are noisy and require further processing to make them suitable for feature extraction. Since existing crowd counting methods use the last convolutional layer for feature extraction, they are, therefore, unable to capture the details of small objects in high-density crowds.

Unlike other methods that use the feature map of the last convolutional layer for crowd counting, we use multiple feature map from different layers to capture the details of small, medium, and large objects. For this purpose, we use fusion strategy of features that fuses the feature maps from the shallow and top layers. We also incorporate the feature maps of mid-level convolutional layers. We assume that utilization of feature maps from different convolutional layers assists the crowd counting task to achieve higher accuracy as possible.

We adopt the fusion strategy adopted in [39] and fuse the feature maps of conv3_*x*, conv4_*x*, and conv5_*x*. The size and number of channels of these feature maps are different. More precisely, the size of feature map of conv5_*x* is 1/2 of the feature map of conv4_*x*. Similarly, the size of feature map of conv4_*x* is 1/2 of the size of conv3_*x*. To effectively fuse feature maps of different resolutions, we perform two steps. First, we need to up-sample the resolution of higher order layers, i.e., conv4_*x*, and conv5_*x* by employing transposed convolutional layer. Second, to make the number of channels of different feature maps equal, we apply 1×1 convolutional layer after transposed convolutional layer.

To combine conv4_*x* and conv5_*x*, we apply transposed convolutional layer of size 2×2 to conv5_*x* to make its size equal to conv4_*x*. Let F_1 represents the fused feature map. We then apply 2×2 transposed convolutional layer to F_1 map to make its size equal to feature map of conv3_*x*. To make the number of channels equal, we apply 1×1 convolutional layer to conv3_*x* with 512 number of channels. We then fuse the feature maps of conv3_*x* and F_1. Let F_2 is the resultant fused feature map. To further suppress the aliasing effect, we apply 1×1 convolution layer to fused feature map. The feature map now combines rich semantic from the deeper layers and also fine-grained information about the small objects from shallow layers. The final feature map is provided as input to prediction layer which employ 1×1 convolution and generates density value for each pixel of the feature map. We then up-sampled the final feature map by employing bi-linear interpolation to make its size equal to the size of input image.

3.1 Training and Implementation Details

We now discuss implementation and training details of the framework. Let $S = \{s_1, s_2, \ldots, s_n\}$ represents n number of training images. We divide each image s_i into patches, each of size l x m. Let $P = \{p_1, p_2, \ldots, p_m\}$, represents the m number of patches involved in training the network. With each patch p_i, we associate a density level d_i, that represents total number

of people in each patch p_i. We randomly choose patches for training and provide then as input to the proposed CNN. We employ a regression method to the learn features that represent crowd count in patch. However, during the training phase, we observed data imbalance problem. This is due to reason that in crowd counting datasets, the ground truth is always provided in the point annotations. Each point corresponds to the location of pedestrian in the scene. Usually, high-density crowds contains few thousands of people. This means that we can generate few thousands of positive samples, while most of pixels will belong to the background. In this way, the number of negative samples will be thousand times greater than positive samples. This creates data imbalance problem which will lead to poor generalization of the crowd counting model.

To address this problem, we employ methods in [5, 55, 63] to generate density map for training the network. Let h_i is the position of pedestrian in the image. Then, delta function $\delta(h - h_i)$ for all positions of pedestrians in the image and can be expressed by Eq. 1

$$H(h) = \sum_{j=1}^{n} \delta(h - h_j), \tag{1}$$

where σ is the variance of the kernel G. The above density function is feasible for the scenes captured from the orthogonal view. Such scenes do not suffer from perspective distortions due to which the size of the analyzed objects is constant. However, these pedestrian crowd scenes do not hold this assumption, where camera is usually installed at tilted position. Such configuration of the camera causes perspective distortions due to which the size of same objects appears different in disparate locations of the scene. To address this problem, we use a Gaussian kernel that compensates perspective distortions to generate density map [63]. To obtain continuous density function, we convolve H with G as in Eq. 2

$$H(h) = \sum_{j=1}^{n} \delta(h - h_j) * G\sigma(h). \tag{2}$$

The sum of the peaks of the density map represents crowd count in the given image. Figure shows the original image and their corresponding true crowd density maps.

We then define the training loss L_E function through which the network learns set of parameters θ. The loss function L_E is the euclidean loss that measure the distance between the true density and predicted density maps and formulated as follows:

$$L_E(\theta) = \frac{1}{N} \sum_{j=1}^{N} || K_d(S_j;\theta) - D_j ||^2, \tag{3}$$

where θ represents the parameter of the network learn during the training process, N represents the number of images used for training, S_j is the current image, and D_j is the ground truth density map of image j. The optimization of Eq. 3 provides high-quality density map that obtains accurate crowd count.

4 Experiment Results

In this section, we evaluate and compare the performance of the proposed framework with other existing methods on three publicly available datasets, namely, UCF_CROWD_50 [29], UCF_QNRF [30], and ShanghaiTech [63] datasets. The proposed network is implemented in the Pytorch framework and trained on NVIDIA TITAN Xp GPU. The experimental setup included 64-bit Ubuntu 16.04, Anaconda 3, CUDA Toolkit 10.2, and Pytorch 1.4.

4.1 Datasets

We provide the details of each dataset as follows:

UCF_CROWD_50 is the first high-density crowd dataset proposed by Idrees et al. [29] for evaluating crowd counting models. The dataset contains 50 images of extremely varying densities and provide 63,974 point annotations. The count in an image ranges from 94 people per image to 4543 people per image and makes an average of 1280 people per image. The images cover different challenging scenes with varying resolutions, camera view points, and backgrounds.

UCF_QNRF dataset is proposed by Idrees et al. [30] and is considered as the most suitable dataset for evaluating crowd counting models. The dataset consists of high-resolution images with diverse background, mainly collected from Web search, Flickr, and Hajj recording archives. The dataset consists of 1535 images covering different scenes of diverse variations in camera view points, illumination, densities, and resolution. The dataset contains 1251,642 point annotations, where each point represent a single head. The dataset is divided into a training and testing sets. The training set contains 1201 images and testing set contains 334 images.

ShanghaiTech dataset is first introduced by Zhang et al. [63]. The dataset contains 1198 annotated images with 330,165 total point annotations. At that time, the dataset was considered as one of the largest due to large number of annotations. The dataset is divided into two parts, i.e., Part A and Part B. There are 482 images in Part A, which are collected from different sources over the Internet. On the other hand, there are 716 images in Part B, which are collected from the busy metropolitan areas of Shanghai. There is significant variation among the densities of two parts. Generally, part A

contains the images with higher densities than part B. This significant variation in crowd densities of two parts poses a challenge for a crowd counting models to accurately estimate the count in images of varying densities. For training and testing, we follow the same convention adopted in [63]. The authors divided part A into 300 training images and the remaining 182 images are used for testing. The training set of Part B contains 400 images, while 316 images are reserved for testing. Figure 4 shows different sample images from each dataset.

4.2 Evaluation Metrics

To quantitatively evaluate the performance of crowd counting models, we use mean absolute error (MAE) and mean square error (MSE) by following the convention adopted in [30, 63]. MAE and MSE are formulated in Eqs. 4 and 5, respectively, as follows:

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|\mu_n - G_n|, \tag{4}$$

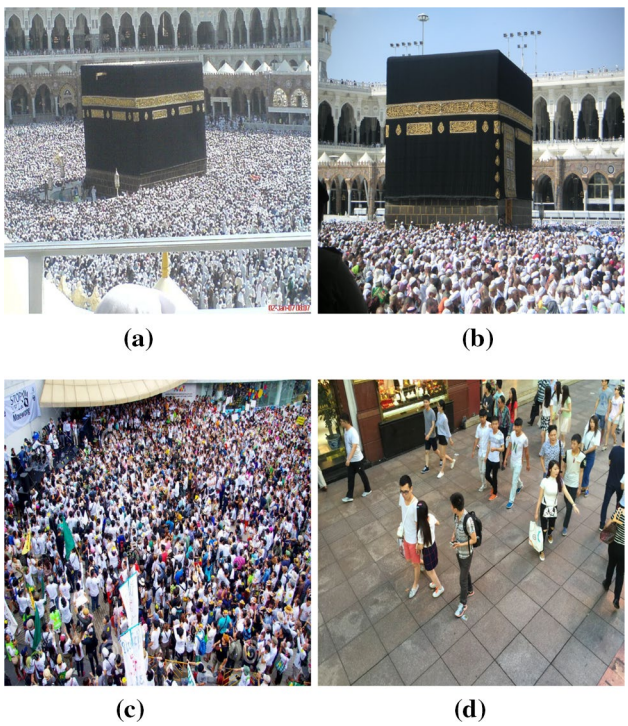


Fig. 4 Sample images of three datasets. **a** Sample image OF UCF_CROWD_50, **b** sample image of UCF_QNRF, **c** sample image of ShanghaiTech part A, and **d** shows the sample image of ShanghaiTech part B

$$MSE = \frac{1}{N}\sum_{n=1}^{N}(\mu_n - G_n)^2, \tag{5}$$

where N represents the number of training images, μ_n is the predicted crowd count, and G_n is ground-truth crowd count of pedestrians at image n. We use the above evaluation metrics to compare proposed approach with other reference methods on all three benchmark datasets.

4.3 Comparison with State-of-the-Art Methods

We compare proposed framework with four most related methods, i.e., Rodriguez et al. [49], Idrees et al. [28], Lempitsky et al. [35], and Zhang et al. [62] on UCF_CROWD_50 dataset. We report quantitative results of each method in Table 1. The table demonstrates that proposed method outperforms other reference methods by producing lowest MAE and MSE scores. From the table, it is obvious that Rodriguez et al. [49] perform lower compare to other methods. This is due to fact that method is detection based and relies on the performance of the detector [20] used in the framework. However, we observe that detection is not a viable solution for crowd counting in high-density crowds. Since, it is challenging to detect heads in high-density crowds due to intra-class variations in scales, appearance, and poses of human heads. Lempitsky et al. [35] learned a density regression model using SIFT features and optimize MESA distance between the predicted density map and ground truth. On the other hand, Zhang et al. [62] achieve comparable values of MAE and MSE, since the authors propose CNN-based crowd counting model that rely on hierarchical features instead of hand-crafted features. However, the proposed model is patch-based and causes much computational complexity during training and testing.

In Table 2, we compare the proposed framework with other reference methods on the ShanghaiTech dataset. From the table, it is obvious that the proposed framework beats other reference methods by achieving low values of MAE and MSE. Chen et al. [10] achieve relatively higher values of MAE and MSE. The method uses traditional Local

Table 1 Evaluation of different methods on UCF_CROWD_50 dataset in terms of MAE and MSE values

Mthod	MAE	MSE
Rodriguez et al. [49]	655.7	697.8
Idrees et al. [28]	468.0	590.3
Lempitsky et al. [35]	493.4	487.1
Zhang et al. [62]	467.0	498.5
Proposed	402.3	434.1

The lower is the better

Table 2 Comparison of different methods on ShanghaiTech part A and part B dataset in terms of MAE and MSE

Method	Part A		Part B	
	MAE	MSE	MAE	MSE
Chen et al. [10]	303.2	371.0	59.1	81.7
Zhang et al. [62]	181.8	277.7	32.0	49.8
Zhang et al. [63]	110.2	173.2	26.4	41.3
Marsden et al. [43]	126.5	173.5	23.76	33.12
Sindagi et al. [55]	101.3	152.4	20.0	31.1
Tang et al. [58]	89.2	141.9	14.7	25.4
Han et al. [24]	79.1	130.1	17.8	26.0
Proposed	77.58	129.7	14.1	21.10

binary pattern (LBP) to extract texture features from the input image and then employ ridge regression to learn the crowd count. This is due to the fact that Chen et al.'s [10] model achieves lower performance, since the model uses LBP features which are blind and cannot distinguish human from the background. Zhang et al. [62] also achieve high values of MAE and MSE relative to other reference methods; however, the method achieves good results compare to Chen et al. [10]. Zhang et al. [63] use multicolumn CNN to solve the multiscale problem and achieve better results than previous approach [62]. Marsden et al. [43] explore fully convolutional network (FCN) for crowd counting. The network takes arbitrary size image and outputs a crowd density map, where high peaks are integrated to produce final crowd count. Sindagi et al. [55] propose a cascaded convolutional network to learn two tasks, i.e., crowd count and density map estimation. The network takes arbitrary size image and outputs a density map. Tang et al. [58] propose fusion CNN that has two key stages. The first stage adopts deep-fusion network to estimate the crowd density and the second stage employs regression to estimate the count. Han et al. [24] left behind proposed framework by a slight margin. The authors adopt divide-and-conquer strategy, and instead of estimating the count from the whole image, they divide the image into multiple overlap patches. Then, from each patch, they hierarchical features are extracted by CNN which is then followed by a fully connected network that regress the count in each patch. Markov random field is then applied to smooth the counting results in adjacent patches.

From Table 2, we further observe that deep learning methods produce better results than the traditional statistical models. However, among deep learning models, our proposed framework achieves best results on both Part_A and Part_B, which highlights the fact that multilayer fusion is effective for accurately estimating the crowd count. Since multilayer fusion combines both high-level semantic information from higher layers and information about the small objects from lower layers. From the table, we further observed that Part_A is more challenging than Part_B, as most of the methods produce higher MAE and MSE values on Part_A than Part_B.

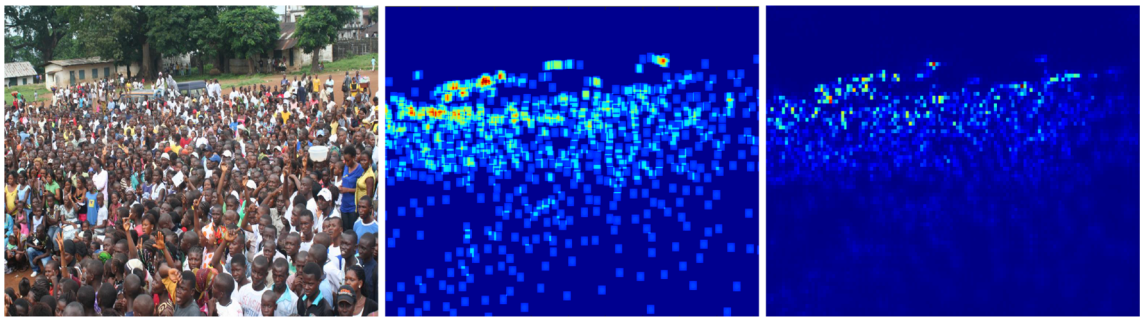
Table 3 Evaluation of different methods on UCF_QNRF dataset using MAE and MSE

Methods	MAE	MSE
Idrees et al. [28]	315.0	508.0
Sindagi et al. [55]	252.0	514.0
Switching CNN [51]	228.0	445.0
Encoder-Decoder [3]	270.0	478.0
MCNN [63]	277.0	426.0
Proposed	218.2	357.4

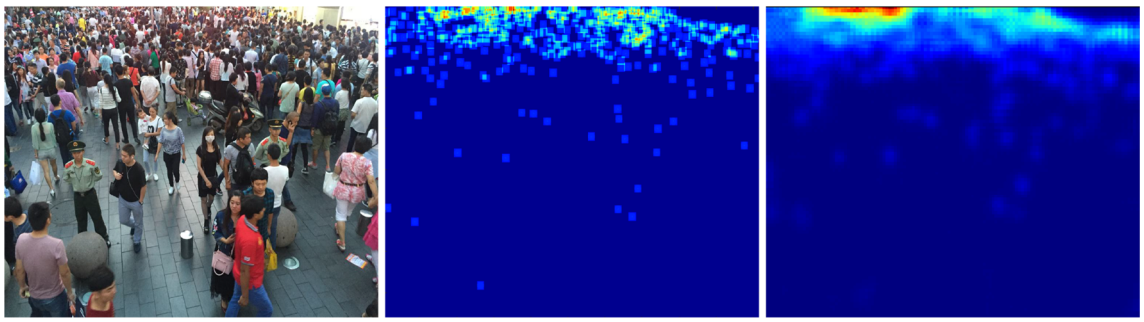
Table 3 shows comparison results of different methods on UCF_QNRF dataset. It is obvious from the table that the proposed method beats other state-of-the-art methods by producing lower values of MAE and MSE. Switching CNN [51] produces comparable results. The method adopts a unique way for handling multiscale variations by leveraging variation of crowd densities in different locations of the input image. The method uses multiple regressors, each of different receptive field to capture scale variations in the image. The switch classifier routes the patches to best CNN regressor based on density level. We also report visualization of the predicted results and corresponding ground truth on three datasets in Figs. 5, 6, and 7.

4.4 Discussion

From experiment results, we observe that distance of camera from the scene and camera view point is the main cause of scale problem. Due to the scale problem, the size of humans near camera appears large than size of human at far distance. To capture such variations in sizes of human heads, it is important to model a network that can handle such scale variations in the image. Zhang et al. [63] propose a CNN network that addresses the scale problem using three column CNN structures. The model produces good results; however, training three column of the network independently causes computational cost. Other reference methods use limited and fixed scale range and, therefore, loss the abilities to learn a generalized model. Furthermore, these methods employ regression techniques to regress the crowd county or crowd density map directly from the image. The performance of these approaches is limited by the following two main reasons: (1) These approaches utilize the feature map of the last convolutional layer that contains rich contextual information about the scenes and, however, do not contain much information about small objects. (2) These approaches use CNNs that consist of subsequent pooling layers that reduce the

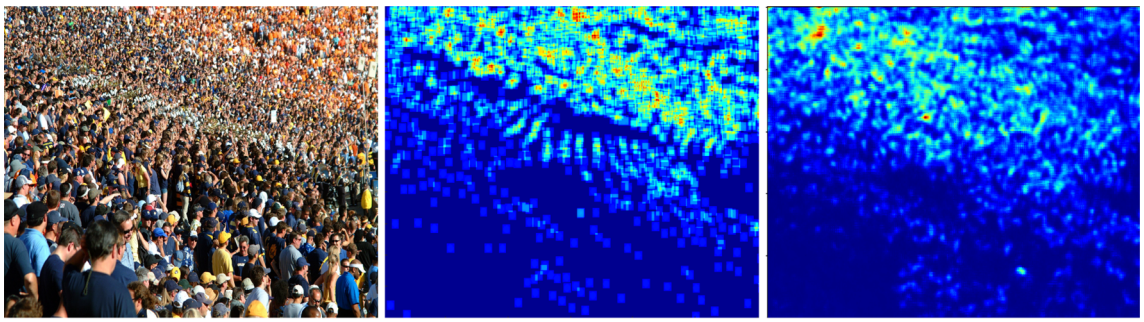


(a) Sample frame from ShanghaiTech Part A dataset **(b)** Ground truth count = 762 **(c)** Predicted count = 724

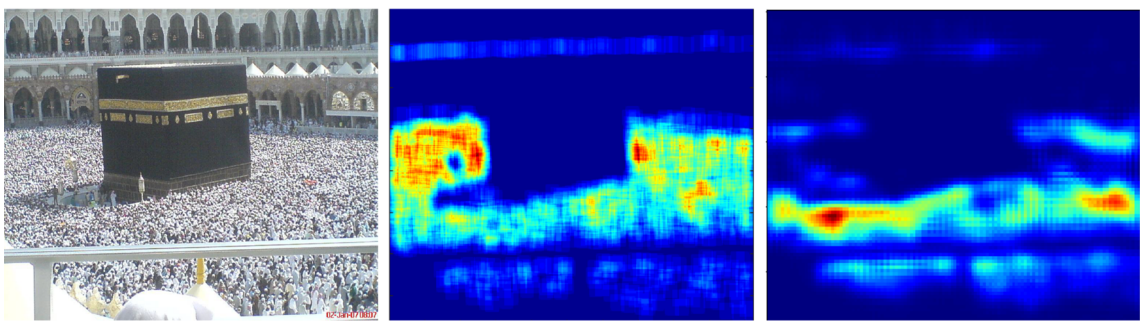


(d) Sample frame from ShanghaiTech Part B dataset **(e)** Ground truth count = 476 **(f)** Predicted count = 437

Fig. 5 Visualization of ground truth and predicted density maps of samples frames selected from ShanghaiTech dataset

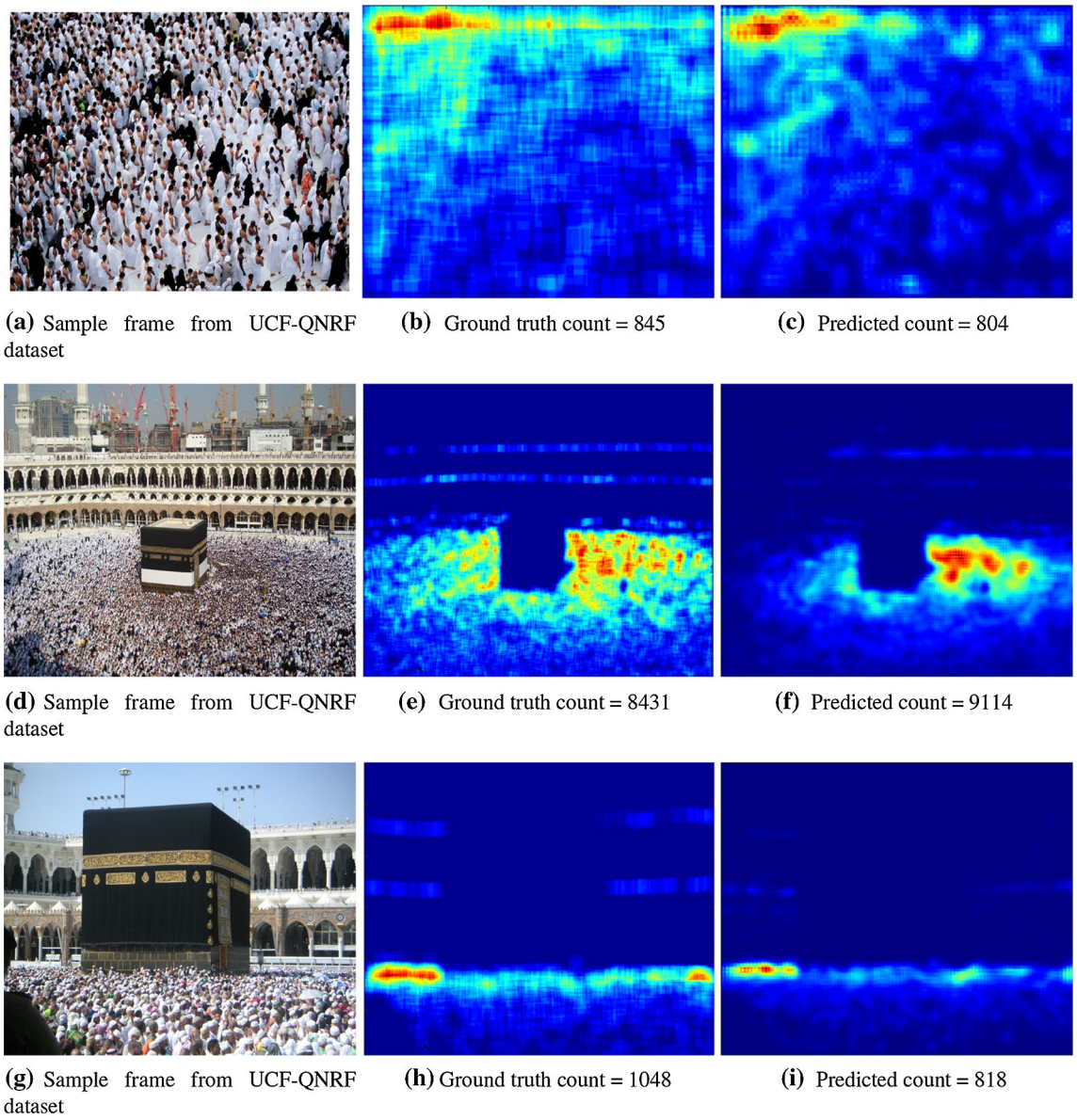


(a) Sample frame from UCF_CC_50 dataset **(b)** Ground truth count = 1940 **(c)** Predicted count = 1970



(d) Sample frame from UCF_CC_50 dataset **(e)** Ground truth count = 2162 **(f)** Predicted count = 2262

Fig. 6 Visualization of ground truth and predicted density maps of samples frames selected from UCF_CC_50 dataset



(a) Sample frame from UCF-QNRF dataset **(b)** Ground truth count = 845 **(c)** Predicted count = 804

(d) Sample frame from UCF-QNRF dataset **(e)** Ground truth count = 8431 **(f)** Predicted count = 9114

(g) Sample frame from UCF-QNRF dataset **(h)** Ground truth count = 1048 **(i)** Predicted count = 818

Fig. 7 Visualization of ground truth and predicted density maps of samples frames selected from UCF-QNRF dataset

resolution of final crowd density map that leads to the loss of crucial information especially in images that consist of high-density crowds with large variations in scales. By contrast, the proposed framework addresses the shortcomings of previous models by adopting affordable and effective way of dealing with scale variations. We assume that higher layers contain rich information about the person near to camera, while lower layers contain information about the person far away from the camera. We fuse the feature map of these multiple layers (of different depths) to adapt the scale variations in the input image. We use the final fused map to learn a mapping function between the heads in image and crowd count. From empirical evidences, we observe that proposed fusion strategy learns multiscale discriminative features and effective to achieve better results compared to other state-of-the-art methods.

5 Conclusion

We presented a deep convolutional neural network that overcomes the problem of scale variations by fusing information from shallow to deep layers. The framework estimated density map and obtained final crowd count by the integration of peaks in density map. We evaluated proposed framework on different publicly available benchmark datasets. From experiment results, we demonstrated the effectiveness of proposed approach. However, we observed the accuracy of

proposed framework still far behind the ground truth. This is due to the fact that in the analyzed high-density crowd datasets, humans are hard to recognize in some images due to low resolution and extremely small size of human head. In future, we plan to propose a network that handles these challenges to achieve high accuracy.

Acknowledgements The work was funded by grant number 14-INF1015-10 from the National Science, Technology, and Innovation Plan (MAARIFAH), the King Abdul-Aziz City for Science and Technology (KACST), Kingdom of Saudi Arabia. We thank the Science and Technology Unit at Umm Al-Qura University for their continued logistics support.

Author Contributions Conceptualization: SDK, BZ, and YS; methodology: SDK; software, SDK; validation: SDK, YS; formal analysis: AN; writing—original draft preparation: SDK, YS, BZ, and AN; writing—review and editing: SDK, YS, BZ, and AN; visualization: SDK and YS; supervision: BZ; project administration: BZ and AN; funding acquisition: AN. All authors have read and agreed to the published version of the manuscript.

Funding The work was funded by Grant Number 14-INF1015-10 from the National Science, Technology, and Innovation Plan (MAARIFAH), the King Abdul-Aziz City for Science and Technology (KACST), Kingdom of Saudi Arabia. We thank the Science and Technology Unit at Umm Al-Qura University for their continued logistics support.

Availability of Data and Materials In this work, we have used publicly available datasets. UCF-QNRF dataset can be downloaded from "https://www.crcv.ucf.edu/data/ucf-qnrf/". ShanghaiTech part_A and part_B can be downloaded from "https://www.kaggle.com/tthien/shanghaitech-with-people-density-map". UCF_CROWD_50 can be downloaded from "https://www.crcv.ucf.edu/data/ucf-cc-50/".

Declarations

Conflict of Interest The authors have no conflict of interests.

References

1. Ali, S., Shah, M.: In: A Lagrangian Particle Dynamics Approach for Crowd Flow Segmentation and Stability Analysis. pp. 1–6. IEEE (2007)
2. Ali, I., Dailey, M.N.: Multiple human tracking in high-density crowds. Image Vis. Comput. **30**(12), 966–977 (2012)
3. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017)
4. Boominathan, L., Kruthiventi, S.S.S., Babu, R.V.: Crowdnet: a deep convolutional network for dense crowd counting. In: Proceedings of the 2016 ACM on Multimedia Conference, MM '16, pp. 640–644, New York, NY, USA, (2016) ACM
5. Boominathan, L., Kruthiventi, S.S.S., Babu, R.V.: Crowdnet: a deep convolutional network for dense crowd counting. Presented at the (2016)
6. Chan, A.B., Morrow, M., Vasconcelos, N.: Analysis of crowded scenes using holistic properties. Presented at the (2009)
7. Chan, A.B., Vasconcelos, N.: In: Bayesian Poisson Regression for Crowd Counting, pp. 545–551. IEEE (2009)
8. Chan, A.B, Liang, Z.-S.J., Vasconcelos, N.: Privacy preserving crowd monitoring: Counting people without people models or tracking. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. pp. 1–7. IEEE (2008)
9. Chan, A.B., Vasconcelos, N.: Counting people with low-level features and bayesian regression. Image Process. IEEE Trans. **21**(4), 2160–2177 (2012)
10. Chen, K., Loy, C.C., Gong, S., Xiang, T.: Feature mining for localised crowd counting. BMVC **1**, 3 (2012)
11. Cho, S.-Y., Chow, T.W.S.: A fast neural learning vision system for crowd estimation at underground stations platform. Neural Process. Lett. **10**(2), 111–120 (1999)
12. Cho, S.-Y., Chow, T.W.S., Leung, C.-T.: A neural-based crowd estimation by hybrid global learning algorithm. Syst. Man Cybernet. Part B: Cybernet. IEEE Trans. **29**(4), 535–541 (1999)
13. Chow, T.W.S., Yam, J.Y.-F., Cho, S.-Y.: Fast training algorithm for feedforward neural networks: application to crowd estimation at underground stations. Artif. Intell. Eng. **13**(3), 301–307 (1999)
14. Conte, D., Foggia, P., Percannella, G., Tufano, F., Vento, M. (eds.): In: A Method for Counting People in Crowded Scenes, pp. 225–232. IEEE (2010)
15. Conte, D., Foggia, P., Percannella, G., Tufano, F., Vento, M. (eds.): In: Counting Moving People in Videos by Salient Points Detection, pp. 1743–1746. IEEE (2010)
16. Dalal, N., Triggs, B.: In: Histograms of Oriented Gradients for Human Detection, vol. 1, pp. 886–893. IEEE (2005)
17. Davies, A.C., Yin, J.H., Velastin, S., et al.: Crowd monitoring using image processing. Electron. Commun. Engi. J. **7**(1), 37–47 (1995)
18. Felemban, E.A., Rehman, F.U., Biabani, S.A.A.,Ahmad, A., Naseer, A., Majid, A.R.M.A.,Hussain, O.K., Qamar, A.M., Felemban, R., Zanjir,F.: Digital revolution for hajj crowd management: a technologysurvey. IEEE Access **8**:208583–208609 (2020)
19. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8. IEEE (2008)
20. 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 (2009)
21. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. Pattern Anal. Mach. Intell. IEEE Trans. **32**(9), 1627–1645 (2010)
22. Fu, M., Xu, P., Li, X., Liu, Q., Ye, M., Zhu, C.: Fast crowd density estimation with convolutional neural networks. Eng. Appl. Artif. Intell. **43**, 81–88 (2015)
23. Girshick, R.: Fast R-CNN. Presented at the (2015)
24. Han, K., Wan, W., Yao, H., Hou, L.: Image crowd counting using convolutional neural network and Markov random field. J. Adv. Comput. Intell. Intell. Inform. **21**(4), 632–638 (2017)
25. Hao, Z., Liu, Y., Qin, H., Yan, J., Li, X., Hu, X. (eds.): Scale-aware face detection. Presented at the (2017)

26. He, Kaiming, Gkioxari, Georgia, Dollár, Piotr, Girshick, Ross: Mask r-cnn. Presented at the (2017)
27. Hidayah, R., Mark, S.N., and John, N.C.: On crowd density estimation for surveillance. 2006
28. Idrees, H., Saleemi, I., Seibert, C., Shah, M.: Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, CVPR '13, pp. 2547–2554, Washington, DC, USA, (2013). IEEE Computer Society
29. Idrees, H., Saleemi, I., Seibert, C., Shah, M.: Multi-source multi-scale counting in extremely dense crowd images. Presented at the (2013)
30. Idrees, H., Tayyab, M., Athrey, K., Zhang, D., Al-Maadeed, S., Rajpoot, N., Shah, M.: Composition loss for counting, density map estimation and localization in dense crowds. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 532–546 (2018)
31. Kang, K., Wang, X.: Fully convolutional neural networks for crowd segmentation. CoRR arXiv:abs/1411.4464, (2014)
32. Khan, S.D., Tayyab, M., Amin, M.K., Nour, A., Basalamah, A., Basalamah, S., Khan, S.A.: Towards a crowd analytic framework for crowd management in majid-al-haram. arXiv preprint. arXiv:1709.05952 (2017)
33. Khan, S.D.: Congestion detection in pedestrian crowds using oscillation in motion trajectories. Eng. Appl. Artif. Intell. **85**, 429–443 (2019)
34. Kong, D., Gray, D., Tao, H.: A viewpoint invariant approach for crowd counting. In: Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, Volume 3, pp. 1187–1190. IEEE (2006)
35. Lempitsky, V., Zisserman, A.: Learning to count objects in images. Adv. Neural Inf. Process. Syst. **23**, 1324–1332 (2010)
36. Liu, J., Gao, C., Meng, D., Hauptmann, A.G.: Decidenet: counting varying density crowds through attention guided detection and density estimation. Presented at the (2018)
37. Liu, Y., Li, Hongyang, Y., Junjie, W., Fangyin, W., Xiaogang, T., Xiaoou, (eds.): Recurrent scale approximation for object detection in CNN. Presented at the (2017)
38. Ma, R., Li, L., Huang, W., Tian, Q.: In: On pixel count based crowd density estimation for visual surveillance, vol. 1, pp. 170–173. IEEE (2004)
39. Ma, C., Mu, X., Sha, D.: Multi-layers feature fusion of convolutional neural network for scene classification of remote sensing. IEEE Access **7**, 121685–121694 (2019)
40. Marana, A.N., Cavenaghi, M.A., Ulson, R.S., Drumond, F.L. (eds.): Real-time crowd density estimation using images. In: Advances in Visual Computing, pp. 355–362. Springer (2005)
41. Marana, A.N., da Fontoura Costa, L., Lotufo, R.A., Velastin, S.A.: Estimating crowd density with minkowski fractal dimension. In: Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on, volume 6, pp. 3521–3524. IEEE (1999)
42. Marana, A.N., Velastin, S.A., Costa, L.F., Lotufo, R.A.: Estimation of crowd density using image processing. In: Image Processing for Security Applications (Digest No.: 1997/074), IEE Colloquium on, pp. 11–1. IET, (1997)
43. Marsden, M., McGuinness, K., Little, S., O'Connor, N.E.: Fully convolutional crowd counting on highly congested scenes. arXiv preprint arXiv:1612.00220 (2016)
44. Marsden, M., McGuinness, K., Little, S., O'Connor, N.E.: Resnetcrowd: In: A residual deep learning architecture for crowd counting, violent behaviour detection and crowd density level classification, pp. 1–7. IEEE (2017)
45. Onoro-Rubio, D., López-Sastre, R.J.: In: Towards Perspective-Free Object Counting with Deep Learning, pp. 615–629. Springer (2016)
46. Ranjan, V., Le, H., Hoai, M.: Iterative crowd counting. Presented at the (2018)
47. Regazzoni, C.S., Tesei, A., Murino, V.: A real-time vision system for crowding monitoring. In: Industrial Electronics, Control, and Instrumentation, 1993. Proceedings of the IECON'93., International Conference on, pp. 1860–1864. IEEE, (1993)
48. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2016)
49. Rodriguez, M., Sivic, J., Laptev, I., Audibert, J.-Y.: Density-aware person detection and tracking in crowds
50. Roqueiro, D., Petrushin, V.A.: Counting people using video cameras. Int. J. Parallel Emerg. Distrib. Syst. **22**(3), 193–209 (2007)
51. Sam, D.B., Surya, S., Babu, R.V.: In: Switching convolutional neural network for crowd counting, pp. 4031–4039. IEEE (2017)
52. Sang, J., Wu, W., Luo, H., Xiang, H., Zhang, Q., Hu, H., Xia, X.: Improved crowd counting method based on scale-adaptive convolutional neural network. IEEE Access **7**, 24411–24419 (2019)
53. Shang, C., Ai, H., Bai, B.: In: End-to-end crowd counting via joint learning local and global count, pp. 1215–1219. IEEE (2016)
54. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
55. Sindagi, V.A., Patel, V.M.: In: Cnn-based Cascaded Multi-Task Learning of High-Level Prior and Density Estimation for Crowd Counting, pp. 1–6. IEEE (2017)
56. Sindagi, V.A., Patel, V.M.: Multi-level bottom-top and top-bottom feature fusion for crowd counting. Presented at the (2019)
57. Solmaz, B., Moore, B.E., Shah, M.: Identifying behaviors in crowd scenes using stability analysis for dynamical systems. IEEE Trans. Pattern Anal. Mach. Intell. **34**(10), 2064–2070 (2012)
58. Tang, S., Pan, Z., Zhou: Xingyu.: Low-rank and sparse based deep-fusion convolutional neural network for crowd counting. Math. Prob. Eng. 2017, (2017)
59. Walach, E., Wolf, L.: In: Learning to count with CNN boosting, pp. 660–676. Springer (2016)
60. Wang, C., Zhang, H., Yang, L., Liu, S., Cao, X.: Deep people counting in extremely dense crowds. In: Proceedings of the 23rd ACM International Conference on Multimedia, pp. 1299–1302 (2015)
61. Xiaohua, L., Lansun, S., Huanqin, L.: Estimation of crowd density based on wavelet and support vector machine. Trans. Inst. Meas. Control **28**(3), 299–308 (2006)
62. Zhang, C., Li, H., Wang, X., Yang: Xiaokang.: Cross-scene crowd counting via deep convolutional neural networks. Presented at the (2015)
63. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: Single-image crowd counting via multi-column convolutional neural network. Presented at the (2016)
64. Zhang, X., Shu, X., He, Z.: Crowd panic state detection using entropy of the distribution of enthalpy. Physica A **525**, 935–945 (2019)
65. Zhou, P., Ni, B., Geng, C., Hu, J., Xu, Y. (eds.): Scale-transferrable object detection. Presented at the (2018)