Hindawi
Mathematical Problems in Engineering
Volume 2017, Article ID 5046727, 11 pages
https://doi.org/10.1155/2017/5046727

Research Article

Low-Rank and Sparse Based Deep-Fusion Convolutional Neural Network for Crowd Counting

Siqi Tang, Zhisong Pan, and Xingyu Zhou

PLA University of Science and Technology, Nanjing, Jiangsu, China

Correspondence should be addressed to Zhisong Pan; hotpzs@hotmail.com

Received 14 March 2017; Revised 10 July 2017; Accepted 25 July 2017; Published 25 September 2017

Academic Editor: Suzanne M. Shontz

This paper proposes an accurate crowd counting method based on convolutional neural network and low-rank and sparse structure. To this end, we firstly propose an effective deep-fusion convolutional neural network to promote the density map regression accuracy. Furthermore, we figure out that most of the existing CNN based crowd counting methods obtain overall counting by direct integral of estimated density map, which limits the accuracy of counting. Instead of direct integral, we adopt a regression method based on low-rank and sparse penalty to promote accuracy of the projection from density map to global counting. Experiments demonstrate the importance of such regression process on promoting the crowd counting performance. The proposed low-rank and sparse based deep-fusion convolutional neural network *(LFCNN)* outperforms existing crowd counting methods and achieves the state-of-the-art performance.

1. Introduction

Recent years have witnessed extensively crowded scenes, such as concerts, political speeches, ceremonies, marathons, and tourist spots. The crowd counting problem, as a machine learning and computer vision problem, takes a single image or a surveillance frame as input and aims to estimate how many people are in it. It is of significant importance to public security and automatic surveillance [1]. Though tremendous strides have been made in crowd counting, it still remains a challenge due to severe occlusion, various perspective distortion, and diverse crowd densities.

To solve these problems and promote the accuracy of crowd counting, much methods have been proposed in the crowd counting literature. This paper is not the first one to leverage convolutional neural network (CNN) model to promote the accuracy of crowd counting, whereas most of the CNN based crowd counting methods adopt a two-stage pipeline: crowd density estimation with an end-to-end deep network and direct integral to obtain the global counting, which accumulates the errors and limits the promotion of counting accuracy. To solve this problem and promote the accuracy of crowd counting, we propose a low-rank and sparse based deep-fusion convolutional neural network *(LFCNN)*, which adopts the low-rank and sparse penalty based regression process instead of the direct integral.

1.1. Contributions. In this paper, we aim to promote the accuracy of crowd counting methods from a single image. Motivated by the density map regression architecture and feature-based global regression architecture, we propose the low-rank and sparse based deep-fusion convolutional neural network for crowd counting, which contains two key components: deep-fusion network for density map regression and a subsequent regression method to map the estimated density map to global counting. As the spatial information and the global counting of crowds are, respectively, used by the two steps, the images are projected step by step, from surveillance frame to gray-scale density map and ultimately to global counting. The contributions of the proposed method can be summarized as follows.

To improve the accuracy of density map regression, inspired by the inception structure of googlenet [2], we propose a deep-fusion network structure to capture multiscale targets in crowded images. In each inception unit of our deep-fusion network, to achieve robustness to variation of peoples size, conv layers with filters of various sizes and numbers are utilized as base networks. At the end of each inception,

the feature maps are concatenated and the intermediate representations of base networks are combined. As a result, the deep-fusion network contains more models with more receptive fields and thus obtains more accurate density map regression result for human of various sizes in surveillance frames. Compared with googlenet, the structure of our fusion network is shallower and simpler, which is more suitable for density map regression task.

Aiming to improve the accuracy of global counting regression, which is the ultimate objective of crowd counting methods, we adopt least squares regression with low-rank and sparse penalty to project the estimated density map to global counting, instead of the direct integral process adopted by most existing CNN based crowd counting methods. The inspiration here is rather intuitive: the estimated density maps are coarse with abundant errors and ambiguity, so it is necessary to build the following regression models to eliminate the errors and obtain overall count. Enlightened by low-rank and sparse learning, which builds upon the theory that signals should contain a low-rank part and a sparse part, we adopt low-rank and sparse penalty on estimated density map. We then solve the problem by transforming the penalty on density map to penalty on regression parameters. Compared with other regression methods, such as Ridge Regression and LSSVM, our proposed method can also be viewed as fine-tuning the estimated density map based on the assumption that an accurate density map should contain low-rank structure and sparse structure.

Experiments on large-scale crowd counting datasets demonstrate that, to our knowledge, *LFCNN* can outperform other methods and achieve the state-of-the-art performance in crowd counting application.

2. Related Works

2.1. Crowd Counting. Existing crowd counting methods can be divided into location-based methods and regression based methods.

The location-based methods are based on the foundation that a crowd is composed of single targets which can be detected and then counted. These methods attempt to locate every single person by detector scanning [3, 4], tracking, and trajectories clustering [5] before getting the counting result. However in extensively crowded scenes, a single person is prone to overlap with another and can hardly be precisely detected, which leads to relatively severe error on counting result.

Another popular crowd counting pipeline, regression based method, treats the whole crowd instead of a single person as target and avoids the challenging task of detecting individual person. These methods, more suitable for extensively crowded scenes, can also be divided into two catalogues, global counting regression methods [6–8] and density map regression methods [9–14].

The global feature-based regression pipeline usually contains three successive steps: (1) foreground segmentation, (2) feature extraction, and (3) crowd counting regression. Pixel features [6], texture features [7], and integrated features [8, 15] were utilized and regression models, such as Gaussian Process [8], Ridge Regression [16], and neural network and random forest [15], were adopted to achieve better performances. Despite the effectiveness of these methods, merely utilizing the global counting as supervision signal without using the spacial information of the crowds largely limited the accuracy of these methods.

Compared with global feature-based regression methods, density map regression methods, proposed by [9], further promoted the accuracy of crowd counting by utilizing the crowd's spatial information contained in density map, which was calculated by the position of each person and denoted the crowd density of a local area. Following this pipeline, [10, 17] promoted the counting accuracy using modified random forest algorithm as regression model of density map.

In recent years, with the prosperity of convolutional neural networks (CNN) in image classification [18, 19], detection [20, 21], segmentation [22], and pedestrian detection [23], the CNN model is also leveraged by crowd counting methods. Zhang et al. [11] firstly proposed CNN based density map regression methods, called *Patch-CNN*, and demonstrated significant improvement on the methods based on hand-crafted features. Based on this pipeline, Zhang et al. [12] adopted three networks with various kernel sizes to construct *MCNN*, which was more adaptive to variations in person size. Inspired by combining the high-level semantic information and low-level detailed features, Boominathan et al. [13] combined deep and shallow networks to construct *Long-short CNN* as density map regression network. Aiming to solve the multiscale problem of person size, Onoro-Rubio and López-Sastre [14] proposed *Hydra-CNN*, using a pyramid of image patches of multiple scales to train multiple networks and benefitting from the integration of multiple models. Another attempt to promote the counting accuracy by model ensemble is the *Boost-CNN* [24], which employed boosting to density map regression CNN model. To sum up, the success of these methods could be attributed to the following two reasons: the automatic learning ability of end-to-end density map regression networks and the usage of spacial information in density maps. These methods attempted to promote the accuracy of density map regression by adopting more and more complicated network structures. However, as the global counting instead of the density map is the objective of counting methods, these methods are exactly not end-to-end trained for global counting regression and there is always a gap between the output of the network and the objective of counting problem. The direct integral, adopted by most of the existing methods to project from density maps to global counting, accumulates the error in estimated density maps and limits the promotion of counting accuracy.

The CNN based counting regression methods, such as *Patch-count CNN* [25], *Patch-multitask CNN* [26], and *TSCCM* [27], applied fully connected layers to directly regress the counting of person in image patches. Though these methods constructed end-to-end network for counting task, a surveillance frame needs to be cut into amounts of patches with each patch counted by the network, which is fairly time-consuming. Moreover, without using the spacial information in density maps, these methods' accuracy is severely limited.

Though lots of crowd counting methods have been proposed as shown above, most existing crowd counting methods, including the CNN based ones, either regress the global counting without the spacial information or utilize direct integral of estimated density maps to obtain the global counting without using the global counting. Adopting CNN based density map estimation architecture and a learning process to project the density map to the overall count, our model differs from the existing methods and benefits from adopting both the spacial information and the global information of crowds.

2.2. Low-Rank and Sparse Structure. Low-rank and sparse structures have been profoundly studied in matrix completion, compressed sensing, and dimensional reduction. Principal Component Analysis (PCA) [28] is based on the assumption that signals usually have low intrinsic complexity, are low-rank, or lie on some low-dimensional manifold. And it operated linear projection to seek such low-rank representation by minimizing the error between the signal and the low-rank representation.

$$\mathbf{X} = \mathbf{L} + \mathbf{E}, \quad \text{rank}\,(\mathbf{L}) \leq r, \tag{1}$$

where $\mathbf{E}$ denotes the noise and $\mathbf{L}$ is the low-rank part of the signal $\mathbf{X}$.

A variant of PCA [28], known as a robust PCA (RPCA) [29, 30], is built upon the theory that signals matrix has low-rank structure and the noise is sparsely distributed, affecting only fraction of the signal matrix entries.

$$\mathbf{X} = \mathbf{L} + \mathbf{E}, \quad \text{rank}\,(\mathbf{L}) \leq r, \ \text{card}\,(\mathbf{E}) \leq k. \tag{2}$$

Furthermore, Go Decomposition (GoDec) [31] proposed low-rank + sparse decomposition of a signal, where

$$\mathbf{X} = \mathbf{L} + \mathbf{S} + \mathbf{E}, \quad \text{rank}\,(\mathbf{L}) \leq r, \ \text{card}\,(\mathbf{S}) \leq k. \tag{3}$$

Chalapathy et al. [32] applied deep neural network to construct robust nonlinear subspace that captures the majority of data points and detect anomaly instances, while allowing for some data to have arbitrary corruption. As the deep network extends the robust PCA model to the nonlinear autoencoder setting, the nonlinearity helped discover potentially more subtle anomalies, which promoted the robustness of the model.

3. Notation and Problem Definition

Notation. We use boldface lowercase letters like $\mathbf{z}$ to denote vectors. Boldface uppercase letters like $\mathbf{Z}$ are used to denote matrices. $\|\|_1$ is used to denote the ℓ_1 norm, and $\|\|_*$ is used to denote the trace norm. $\odot$ denotes the Hadamard products.

Problem Definition. Suppose that we have N images, denoted with $\chi = \{(\mathbf{I}_i, h_i)\}_{i=1}^{N}$, where $\mathbf{I}_i$ is the ith image and h_i is the number of people in this image. Person in this image is denoted with $\tau = \{t_j\}_{j=1}^{h_i}$ and the location of person t_i is (x_j, y_j). The density map is denoted by $\mathbf{D}$. More specifically, $\mathbf{D}_i$ denotes the density map calculated as ground truth of density map regression network, $\mathbf{D}_i'$ denotes the density map estimated by CNN networks, and $\mathbf{D}_i''$ denotes the density map modified by the low-rank and sparse regression method.

4. Deep-Fusion Density Map Regression Network

In this section, we illustrate the proposed deep-fusion network structure for crowd density map regression. The goal of the deep-fusion network is to learn a density map regression function $F = \mathbf{I} \rightarrow \mathbf{D}$, where $\mathbf{I}$ is the surveillance image of arbitrary scene and $\mathbf{D}$ is the crowd density map of it. So we firstly illustrate the calculation method of the supervision signal $\mathbf{D}$, based on which we further explain the deep-fusion network structure with the some detail of the network.

4.1. Density Map Calculation. The first step is to calculate the density maps as the training ground truth of the network. With position of each pedestrian labeled, the true density map is actually decided by the pedestrians location, shape, and perspective distortion. Due to severe occlusions, pedestrians' bodies overlap with each other and head is the main cue to judge whether there exists a pedestrian in extensively crowded images. So our work follows [9] and adopts the Gaussian kernel centered on the locations of pedestrians head to denote each pedestrian in the calculated density map, as in

$$\mathbf{G}\left(t_j\right) = \frac{1}{2\pi\sigma^2} e^{((x-x_j)^2+(y-y_j)^2)/2\sigma^2}, \tag{4}$$

where t_j is the jth pedestrian and (x_j, y_j) is its location. Actually, the parameter of Gaussian kernel should correlate with the size of each head, which is influenced by the height and angle of the surveillance camera according to the perspective distortion theory. Most of the scene-specific methods [8] get the parameter by measuring the perspective distortion parameter of each scene as prior knowledge of crowd counting model. However, for arbitrary scene, measuring every single image to get its perspective distortion parameter is much too time-consuming and almost impossible. In our model, we define a global constant parameter for all training images based on the average size of all the heads in datasets.

The density map of the whole image is calculated as a sum of Gaussian kernels of all the pedestrians as in

$$\mathbf{D}_i = \sum \delta\left(x - x_j, y - y_j\right) \mathbf{G}_\sigma\left(t_j\right), \tag{5}$$

where $\mathbf{D}_i$ is the calculated density map of $\mathbf{I}_i$ and $\delta(x - x_j, y - y_j)$ stands for the impulse function.

4.2. Deep-Fusion Network. Multiscale is a significant problem of almost all current computer vision tasks, especially in crowd counting problems owing to the perspective distortion of surveillance cameras [12]. Motivated by googlenet [2], as shown in Figure 1, which integrates several paratactic conv layers with various perspective fields in a inception unit, we propose to use a deep-fusion network to manipulate the scale variation problem. The overall structure of our deep-fusion network is illustrated in Figure 2.

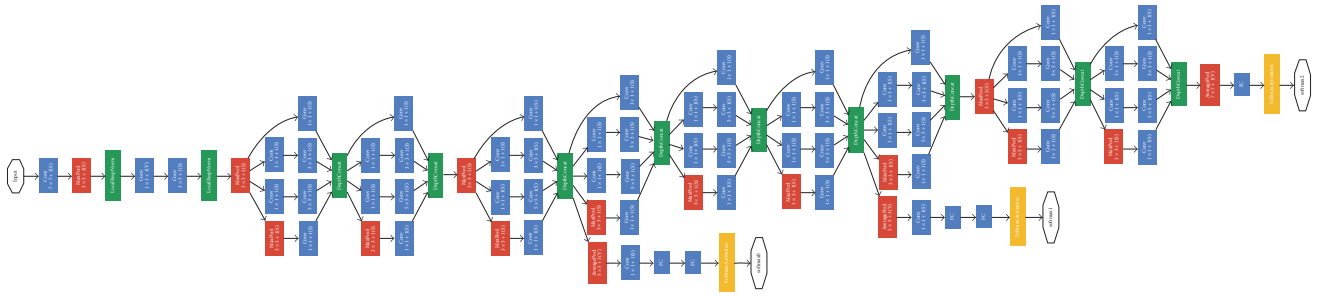


Figure 1: Network structure of googlenet.

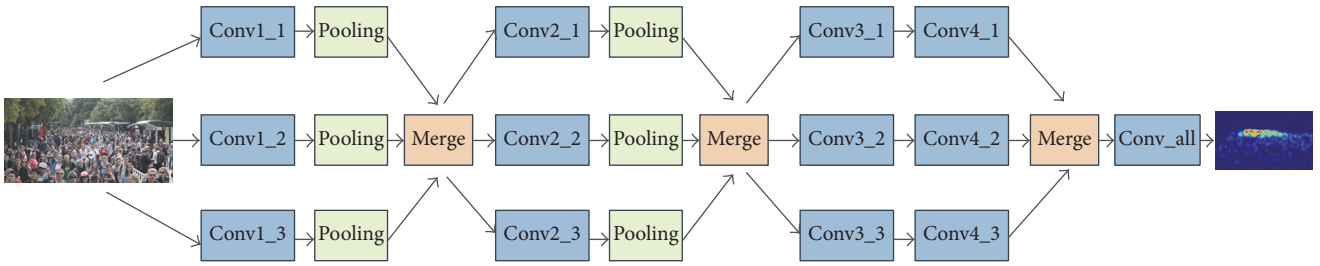


Figure 2: Deep-fusion network structure.

The fusion unit, constructed by three paratactic base networks, is the basic cell of the proposed deep-fusion network. At the end of each fusion unit, the feature maps of base networks are concatenated to obtain the fused representation, which serves as the input of the next fusion unit.

To further illustrate the scale-invariance property of this deep-fusion network structure, we use k_p^q to denote the nonlinear function of pth base network in qth fusion unit and $\mathbf{R}_q$ to denote the output representation of the qth fusion unit; then we have

$$\mathbf{R}_q = F_q\left(k_q^1\left(\mathbf{R}_{q-1}\right), k_q^1\left(\mathbf{R}_{q-1}\right), k_q^1\left(\mathbf{R}_{q-1}\right)\right), \tag{6}$$

where F_q denotes the fusing function of the qth inception and we adopt concatenating as fusing function in this paper.

Through analyzing the information flow path, we can figure out that the output representation of a fusion unit comes from the three base networks and can also flow to the next three base networks of the next fusion unit. Specifically, in our network structure, the three base networks in the first fusion unit bring about three information flow paths of the whole network, $k_1^1 \to k_2^1 \to k_3^1$, $k_1^2 \to k_2^1 \to k_3^1$, and $k_1^3 \to k_2^1 \to k_3^1$. That is, with three fusion units, our deep-fusion network actually contains $3^3 = 27$ information flow paths and each path corresponds to a latent network with specific receptive field, which can capture targets of a specific range of size. As a result, the fused network integrates 27 latent networks and is able to capture heads of various sizes.

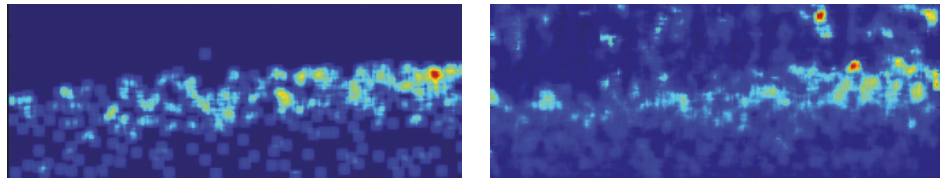
As the heads are small instances, the receptive field of the network should be considerably small to match it and more detailed information instead of the semantic information should be adopted, compared with conventional networks of classification task and detection task. Owing to the fact that the receptive field is enlarged and detailed features are projected to semantic features layer by layer with the growing of the network's depth, the proposed deep-fusion network is rather shallow, compared with resnet and googlenet. The configuration of our network layers is shown in Table 1.

Table 1: Configuration of the conv layers in deep-fusion network.

Layer	Configuration
Conv 1_1	Filter $16 \times 9 \times 9$, pad 4, Relu, pool 2×2
Conv 1_2	Filter $24 \times 7 \times 7$, pad 3, Relu, pool 2×2
Conv 1_3	Filter $32 \times 5 \times 5$, pad 3, Relu, pool 2×2
Conv 2_1	Filter $32 \times 7 \times 7$, pad 3, Relu, pool 2×2
Conv 2_2	Filter $48 \times 5 \times 5$, pad 2, Relu, pool 2×2
Conv 2_3	Filter $64 \times 3 \times 3$, pad 1, Relu, pool 2×2
Conv 3_1	Filter $16 \times 7 \times 7$, pad 3, Relu
Conv 3_2	Filter $24 \times 5 \times 5$, pad 2, Relu
Conv 3_3	Filter $32 \times 3 \times 3$, pad 1, Relu
Conv 4_1	Filter $8 \times 7 \times 7$, pad 3, Relu
Conv 4_2	Filter $12 \times 5 \times 5$, pad 2, Relu
Conv 4_3	Filter $16 \times 3 \times 3$, pad 1, Relu
Conv_all	Filter $1 \times 1 \times 1$, pad 0

What is more, rectified linear unit (ReLU) [33] is adopted as activation function and Max pooling is used. The fully convolutional layer, instead of the fully connected layer, is used to estimate the density map based on the concatenated feature maps, which not only largely reduces the size of parameters but enables the input image to be of arbitrary size as well. At length, the Euclidean loss is defined as the loss function of our deep-fusion network, as illustrated in

$$L = \frac{1}{2N}\sum_{i=1}^{N}\left\|F\left(I_i\right) - D_i'\right\|_2^2. \tag{7}$$



(a) (b)

Figure 3: Calculated density map (a) and estimated density map (b) of 132nd test image in Part A dataset.

5. Low-Rank and Sparse Based Regression

In this section, we further illustrate the motivation, problem definition, solution, and some comments of our method's second component, low-rank and sparse regression from density map to overall counting.

5.1. Motivations. The output of density map regression network is the two-dimensional crowd density map, while the ultimate objective of crowd counting methods is the counting of persons. To our knowledge, most of the existing CNN based methods simply sum up all values of the estimated density map as overall counting, which accumulates the error in estimated density map without using the global counting information of the image. To solve this problem, in this paper, we propose for the first time to use a subsequent regression method to project the estimated density map to the final crowd counting. The inspirations are listed below.

Inspired by the feature-based methods, which use handcraft features and regression methods to obtain the overall crowd count, it is intuitive to employ the feature maps of conv networks as extracted features of subsequent counting regression model to project the estimated density map to the overall crowd counting.

In comparison with the density maps calculated in (5), the estimated density maps are more coarse and noisy, containing much more small nonzero values in background areas, as shown in Figure 3. The cause of the small nonzero values is that the background objects, such as buildings, sky, and tress, are mistreated as human targets by the density map regression network. Such noise and estimation errors are accumulated by direct integral in most of the existing CNN based methods, which limit the accuracy of counting. To eliminate such errors, enlightened by GoDec [31], we leverage the low-rank and sparse penalty on the estimated density map while regressing global counting. In other words, this regression can also be regarded as a modification process to construct the more accurate density maps with low-rank and sparse structure from density maps estimated by CNN network.

5.2. Definition of Counting Regression. On the one hand, the global counting regression problem is essentially the projection from a high dimensional feature to a number; we can formulate it with the commonly used regression function in

$$\min_{\mathbf{w}} \quad \|\mathbf{X}\mathbf{w} - \mathbf{h}\|_2^2, \tag{8}$$

where $\mathbf{h} = [h_1, \ldots, h_i, \ldots, h_N]^T$ and h_i is the number of labeled pedestrians in image $\mathbf{I}_i$. We also represent the $m \times n$ gray-scale density map $\mathbf{D}_i'$ by the vector $\mathbf{x}_i \in R^v$ ($v = mn$) by concatenating its columns; therefore the matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots \mathbf{x}_N] \in R^{v \times N}$ contains the N density maps estimated by the deep-fusion network.

Thus, $\mathbf{w} = [w_1, \ldots, w_i, \ldots, w_v] \in R^v$ is the parameter to map the density map to the overall count. The global counting of the ith image can be calculated by

$$\widehat{h}_i = \mathbf{x}_i \mathbf{w}. \tag{9}$$

On the other hand, in the context of density map modification, the error of overall counting is caused by the errors of the coarse estimated density map. Therefore, the global regression problem can also be viewed as illuminating the errors and constructing the more accurate density map with low-rank and sparse structure. Consequently, $\mathbf{W} \in R^{m \times n}$, which is reshaped by $\mathbf{w}$, denotes the learnt parameter to eliminate the noise of the estimated density map, where the modified density map $\mathbf{D}_i''$ can be calculated by the estimated density map $\mathbf{D}_i'$ with

$$\mathbf{D}_i'' = \mathbf{D}_i' \odot \mathbf{W} = \left[d_{ij}' w_{ij}\right], \tag{10}$$

where $\odot$ denotes the Hadamard products and the entry in the ith raw and jth column of $\mathbf{D}_i''$, d_{ij}'', is the product of d_{ij}' and w_{ij}, which are the entries in the same position of the two matrices. Thus the overall crowd counting is the integral of the modified density map:

$$\widehat{h}_i = \sum_{i=1}^{n}\sum_{j=1}^{m} d_{ij}'' = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} d_{ij}' = \mathbf{1}^T \left(\mathbf{D}_i' \odot \mathbf{W}\right)\mathbf{1}, \tag{11}$$

where $\mathbf{1}$ is unit vector. So the counting regression problem can also be defined in

$$\min_{\mathbf{W}} \quad \sum_{i=1}^{N} \left(\mathbf{1}^T \left(\mathbf{D}_i' \odot \mathbf{W}\right)\mathbf{1} - h_i\right)^2. \tag{12}$$

5.3. Low-Rank and Sparse Penalty. In Section 5.2, we define the basic global counting regression problem in two prospects: regression model and density map modification model. Moreover, the theory of RPCA and GoDec [31] illustrates that signals should contain low-rank and sparse structure, as shown in

$$\begin{aligned} \min_{\mathbf{L},\mathbf{S}} \quad & \|\mathbf{X}-\mathbf{L}-\mathbf{S}\|_F^2 \\ \text{s.t.} \quad & \operatorname{rank}(\mathbf{L}) \le r \\ & \operatorname{card}(\mathbf{S}) \le k. \end{aligned} \tag{13}$$

As the density map is two-dimensional signal and should contain low-rank and sparse structure, enlightened by GoDec, we attempt to add low-rank and sparse penalty on the modified density map to eliminate the errors.

$$\begin{aligned} \min_{\mathbf{L}_i,\mathbf{S}_i} \quad & \sum \left\|\mathbf{D}_i'' - \mathbf{L}_i - \mathbf{S}_i\right\|_F^2 \\ \text{s.t.} \quad & \operatorname{rank}(\mathbf{L}_i) \le r \\ & \operatorname{card}(\mathbf{S}_i) \le k, \end{aligned} \tag{14}$$

where the modified density map is calculated by $\mathbf{D}_i'' = \mathbf{D}_i' \odot \mathbf{W}$ as in Section 5.2 and $\mathbf{L}_i$ and $\mathbf{S}_i$ are the ith modified density map's low-rank and sparse structure, respectively. The target of this optimization problem actually is $\{\mathbf{L}_i, \mathbf{S}_i\}_{i=1}^N$ and all the low-rank structures and sparse structures of density maps need to be counted with the counting as supervision signal, which is difficult to solve.

To solve this problem, as the estimated density map $\mathbf{D}_i'$ is a constant matrix with a constant rank($\mathbf{D}_i'$) and Horn [34] has theoretically justified rank($\mathbf{A} \odot \mathbf{B}$) ≤ rank($\mathbf{A}$)rank($\mathbf{B}$), the low-rank constrain of $\mathbf{L}_i''$ can be relaxed to the constraint that parameter matrix $\mathbf{W}$ contains a low-rank part $\mathbf{L}_\mathbf{W}$. Moreover, as the density map $\mathbf{D}_i'$ is also a dense matrix with abundance of nonzero values, $\mathbf{W}$ should also contain a sparse part $\mathbf{S}_\mathbf{W}$ to ensure $\mathbf{S}_i = \mathbf{D}_i' \odot \mathbf{S}_\mathbf{W}$ is sparse for all density maps. To sum up, the low-rank and sparse penalty on modified density maps can be transformed to the weight matrix $\mathbf{W}$, as shown in

$$\mathbf{W} = \mathbf{L}_\mathbf{W} + \mathbf{S}_\mathbf{W}. \tag{15}$$

So the previous optimization problem can be formulated as

$$\begin{aligned} \min_{\mathbf{L}_\mathbf{W},\mathbf{S}_\mathbf{W}} \quad & \left\|\mathbf{W}-\mathbf{L}_\mathbf{W}-\mathbf{S}_\mathbf{W}\right\|_F^2 \\ \text{s.t.} \quad & \operatorname{rank}(\mathbf{L}_\mathbf{W}) \le r \\ & \operatorname{card}(\mathbf{S}_\mathbf{W}) \le k. \end{aligned} \tag{16}$$

Nevertheless, GoDec is mainly a signal decomposition method used for matrix completion and background modeling by capturing the main part of the signal without supervision, as shown in (2) and the problem in (3) is mainly a matrix construction problem instead of the regression problem as shown in Section 5.2.

The problem defined in Section 5.2 is indeed a regression problem with global counting as supervision signal, whose density map is constrained by the low-rank and sparse structure. To solve this problem, we add the low-rank and sparse penalty of regression weight matrix on density map regression problem formulated in (12). Thus the previous regression problem becomes

$$\begin{aligned} \min_{\mathbf{W}} \quad & \sum_{i=1}^{N} \left(\mathbf{1}^T \left(\mathbf{D}_i' \odot \mathbf{W}\right)\mathbf{1} - h_i\right)^2 \\ \text{s.t.} \quad & \operatorname{rank}(\mathbf{L}_\mathbf{W}) \le r \\ & \operatorname{card}(\mathbf{S}_\mathbf{W}) \le k \end{aligned} \tag{17}$$

which equals

$$\begin{aligned} \min_{\mathbf{L}_\mathbf{W},\mathbf{S}_\mathbf{W}} \quad & \sum_{i=1}^{N} \left(\mathbf{1}^T \left(\mathbf{D}_i' \odot \left(\mathbf{L}_\mathbf{W} + \mathbf{S}_\mathbf{W}\right)\right)\mathbf{1} - h_i\right)^2 \\ \text{s.t.} \quad & \operatorname{rank}(\mathbf{L}_\mathbf{W}) \le r \\ & \operatorname{card}(\mathbf{S}_\mathbf{W}) \le k. \end{aligned} \tag{18}$$

5.4. Solution. To solve this problem, we transform the problem mathematically to

$$\begin{aligned} \min_{\mathbf{L}_\mathbf{W},\mathbf{S}_\mathbf{W}} \quad & L \\ & = \sum_{i=1}^{N} \left(\mathbf{1}^T \left(\left(\mathbf{L}_\mathbf{W} + \mathbf{S}_\mathbf{W}\right) \odot \mathbf{D}_i'\right)\mathbf{1}, h_i\right)^2 \\ & \quad + \alpha \left\|\mathbf{L}_\mathbf{W}\right\|_* + \beta \left\|\mathbf{S}_\mathbf{W}\right\|_1, \end{aligned} \tag{19}$$

where the trace norm regularization term encourages the desirable low-rank structure in the matrix $\mathbf{L}_\mathbf{W}$, the ℓ_1-norm regularization term induces the desirable sparse structure in the matrix $\mathbf{S}_\mathbf{W}$, and α and β are nonnegative trade-off parameters. When the matrices $\mathbf{L}_\mathbf{W}$ and $\mathbf{S}_\mathbf{W}$ are resized to vectors $\mathbf{l}_\mathbf{W}$ and $\mathbf{s}_\mathbf{W}$, the vector version of $\mathbf{W}$, $\mathbf{w}$, equals $\mathbf{l}_\mathbf{W} + \mathbf{s}_\mathbf{W}$ and the whole problem can be represented as

$$\min_{\mathbf{L}_\mathbf{W},\mathbf{S}_\mathbf{W}} \quad \left\|\mathbf{X}^T\left(\mathbf{l}_\mathbf{W} + \mathbf{s}_\mathbf{W}\right), \mathbf{h}\right\|_2^2 + \alpha \left\|\mathbf{L}_\mathbf{W}\right\|_* + \beta \left\|\mathbf{S}_\mathbf{W}\right\|_1. \tag{20}$$

Note that we can solve this optimization problem with [35].

5.5. Comments. This low-rank and sparse based regression method is actually a regression projection from feature to number, with the low-rank and sparse penalty on modified density map. To solve the problem, we transform the penalty on density map to regression parameter matrix.

While our model is not a popular end-to-end architecture, the following process of low-rank and sparse learning can also be viewed as another pairwise product layer, whose weight is trained with low-rank and sparse penalty. Zhang et al. [11] also attempt to add a fully connected layer, without penalty, after the estimated density map to solve the gap between density map and the overall count, but the performance of the network degenerates a lot, which is partly owing to the reason that the overfit problem harms the convergence of the network. By analyzing the essence of the crowd counting problem, we attribute the reason of our models success to the penalty of parameter, which correlates the density map's low-rank and sparse structure of this task.

6. Experiments

Experiments show that our model can promote the accuracy and robustness of the existing crowd counting methods. Implementation of the proposed model is based on the Caffe framework developed by [36] and MALSAR toolbox released by [37].

6.1. Evaluation Metrics. Following the existing works [8, 11, 12], we evaluate the accuracy of each counting method with mean absolute error (MAE), mean squared error (MSE), and mean relative error (MRE).

$$\begin{aligned} \text{MAE} &= \frac{1}{N}\sum_{i=1}^{N}\left|h_i - \widehat{h}_i\right|, \\ \text{MSE} &= \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(h_i - \widehat{h}_i\right)^2}, \\ \text{MRE} &= \frac{1}{N}\sum_{i=1}^{N}\frac{\left|h_i - \widehat{h}_i\right|}{h_i}. \end{aligned} \tag{21}$$

6.2. Datasets. We evaluate our *LFCNN* model on two existing large-scale datasets, Shanghaitech dataset and WorldExpo10 dataset, instead of the low-density or single-scene datasets, such as USCD [38] and Pest2009 [39].

Shanghaitech is a large-scale crowd counting dataset released in 2016 [12], which contains 1198 annotated images with 330165 people located in the centers of their heads. There are two parts in it. Part A contains pictures captured through the Internet with arbitrary size and scene, while Part B is composed of frames collected by several different surveillance cameras in a crowded street.

WorldExpo10 dataset [11] was another existing large-scale crowd counting dataset containing 1132 annotated video sequences which are captured by 108 surveillance cameras in the campus of WorldExpo. Different from the cross scene crowd counting application of [11], we divide all of the annotated images to train datasets and test datasets with the proportion of 6 : 4 to evaluate the methods' performance in arbitrary scene crowd counting.

6.3. Experiment Details. Following the data augmentation method of [12], we crop 9 patches from each training image. As the proposed deep-fusion network has pooling layers, the height and width of the output density maps are a quarter of the input images. So the used supervision density map is reshaped from calculating the density map, where the integral of the values still equals the number of persons for each image. We compare our model with methods of three catalogues of crowd counting architectures. These methods are listed as follows:

(i) Location-based method including *ACF* (Aggregate Channel Feature) [3]

(ii) Feature-based regression methods, including *LBP* [40] combined with Ridge Regression, *LBP* with *LSSVM* [41] regression, and *Gabor* [42] with *LSSVM* regression

(iii) CNN based density map regression methods, including *Patch-count CNN* [25], *Patch-multitask CNN* [26], *TSCCM* [27], *MCNN* [12], *Long-short CNN* [13], and *Hydra-CNN* [14].

Table 2: Performance comparison of crowd counting methods for Shanghaitech Part A dataset.

Network	Part A dataset		
	MAE	MSE	MRE
ACF [3]	390.5	526.8	84.7
LBP + RR	303.2	371.0	70.4
LBP + LSSVM	224.5	294.6	83.3
Patch-CNN [11]	179.7	252.9	67.7
MCNN [12]	110.2	173.2	37.9
Patch-count CNN [25]	118.4	171.1	39.4
Patch-multitask CNN [26]	110.1	170.1	37.0
TSCCM [27]	115.8	167.9	38.5
Long-short CNN [13]	99.1	145.3	30.0
Hydra-CNN [14]	95.7	143.2	26.1
Deep-fusion network	97.9	145.1	29.5
Fusion + RR	126.9	187.8	31.7
Fusion + LSSVM	99.3	145.2	29.1
LFCNN	**89.2**	**141.9**	**17.3**

6.4. Accuracy of Crowd Counting. In Tables 2, 3, and 4, we compare the performance of the proposed *LFCNN* method with those of other existing crowd counting methods of three catalogues on Shanghaitech Part A dataset, Shanghaitech Part B dataset, and WorldExpo10 dataset, respectively. In addition to the low-rank and sparse based regression method, we also adopt other regression methods, such as Ridge Regression and *LSSVM*, to evaluate the effect of low-rank and sparse penalty.

Among these methods, location-based method, *ACF*, cannot achieve decent crowd counting accuracy, which is largely due to the severe occlusion in extensively crowded scenes, especially the Part A dataset. The effect of the feature-based regression methods largely depends on the type of hand-crafted features and regression methods. In addition, it is not surprising to find that the CNN based methods enjoy the preferable accuracy.

Clearly, experimental results demonstrate that the proposed *LFCNN* method outperforms all existing CNN based crowd counting methods with a margin and reduces the MRE (mean relative error) of state-of-the-art methods by 33.71%, 23.87%, and 19.80%, respectively, on three large-scale datasets. The Shanghaitech Part A dataset is of highest average crowd counting and the proposed *LFCNN* shows highest accuracy promotion on it, which demonstrates our proposed method's preferable performance on extensively crowded scenes.

One interesting issue we observe is that the performance of deep-fusion network with Ridge Regression degenerates a lot compared with the direct integral, which might be caused by the fact that the ℓ_1-norm penalty of the Ridge

Table 3: Performance comparison of crowd counting methods for Shanghaitech Part B dataset.

Network	Part B dataset		
	MAE	MSE	MRE
ACF [3]	69.7	108.0	70.4
LBP + RR	59.1	81.7	69.2
LBP + LSSVM	48.3	67.8	57.6
Patch-CNN [11]	32.0	49.8	37.6
MCNN [12]	26.4	41.3	24.2
Patch-count CNN [25]	26.1	37.7	25.9
Patch-multitask CNN [26]	20.3	31.0	22.6
TSCCM [27]	21.7	32.4	20.26
Long-short CNN [13]	19.8	33.1	18.1
Hydra-CNN [14]	17.1	26.3	15.5
Deep-fusion network	17.3	28.9	16.4
Fusion + RR	20.5	28.7	18.9
Fusion + LSSVM	17.6	30.1	15.3
LFCNN	**14.7**	**25.4**	**11.8**

Table 4: Performance comparison of crowd counting methods for the WorldExpo10 dataset.

Network	The WorldExpo10 dataset		
	MAE	MSE	MRE
ACF [3]	41.79	52.36	79.56
LBP + RR	31.01	44.53	80.97
LBP + LSSVM	28.86	42.79	74.69
Gabor + LSSVM	33.61	46.69	84.53
Patch-CNN [11]	12.90	9.62	40.96
MCNN [12]	11.60	16.78	36.50
Patch-count CNN [25]	12.56	17.75	35.21
Patch-multitask CNN [26]	10.56	14.86	30.76
TSCCM [27]	13.18	18.76	36.38
Long-short CNN [13]	13.93	19.70	41.71
Hydra-CNN [14]	8.76	11.83	25.25
Deep-fusion network	10.48	15.04	28.99
Fusion + RR	30.43	41.17	152.28
Fusion + LSSVM	13.81	16.60	67.59
LFCNN	**7.78**	**11.57**	**20.25**

Regression method does not correlate with counting regression problem. Deep-fusion with *LSSVM* can only achieve comparable accuracy with the direct integral strategy, which further demonstrates the crucial role low-rank and sparse penalty plays in global counting regression.

In Figure 3, we show the density map regression and crowd counting result of some test images. The first column is test image, the second column contains the ground truth density maps calculated by the annotated targets, the density maps in the third and fourth column are calculated by *Patch-CNN* [11] and *MCNN* [12], respectively, the fifth column ones are calculated by our proposed deep-fusion network, and the ones in the sixth column are the modified density maps calculated by the point production of weight matrix of low-rank and sparse regression and the density maps in the fifth column.

Table 5: Performance of various network structures on Shanghaitech Part A dataset.

Network	Part A dataset		
	MAE	MSE	MRE
One-column network	179.7	252.9	67.7
Three-column network	110.2	173.2	37.9
Alexnet_density	125.1	185.4	41.6
VGGnet_density	128.5	189.6	43.5
Deep-fusion	**97.9**	**145.1**	**29.5**

Compared with the ground truth density maps, the estimated ones calculated by density regression networks are more coarse and contain much more nonzero small errors in background areas. Among the estimated density maps, the modified ones calculated by our proposed *LFCNN* are more accurate and fine-grained. By comparing the fifth column and the sixth column, we can figure out that the accuracy is largely due to the low-rank and sparse regression process.

6.5. Deep-Fusion Network. To illustrate the density map regression performance with respect to network structure, we construct *MCNN* network structure following [12] and density map regression network based on Alexnet [43] and VGGnet [44] by replacing the fully connected layers with fully convolutional layers. The VGGnet contains 16 layers, which is much too deep for density map regression task as the gradient back-propagated from the density map is considerably small and vanishes in a deep network structure. So we choose the first 6 layers of VGGnet to construct the VGGnet_density network. The density map regression performance of the networks on Shanghaitech Part A dataset is shown in Table 5.

The crowd counting performance of various network structures shown in Table 5 verifies the accuracy of deep-fusion structure on density map regression task. The proposed deep-fusion network structure reduces the MAE metrics by 45.58%, 11.16%, 21.74%, and 23.81% compared with one-column network, three-column network, Alexnet-based density map regression network, and VGGnet-based density map regression network. The effect of the proposed deep-fusion network can be attributed to its capability of capturing multiscale small targets, which is one of the key problems in surveillance based crowd counting. As only pedestrian needs to be detected in crowd counting problem instead of the thousands of objects in image classification task, the abundant features extracted by Alexnet and VGGnet may not be able to show their superb capability. Tables 2, 3, and 4 also demonstrate that the accuracy of crowd counting is enhanced by two aspects, deep-fusion network structure and the low-rank and sparse based regression process.

6.6. Robustness of Crowd Counting. To compare the robustness characteristics of the methods and analyze the scale-invariant capability of the CNN based methods in more

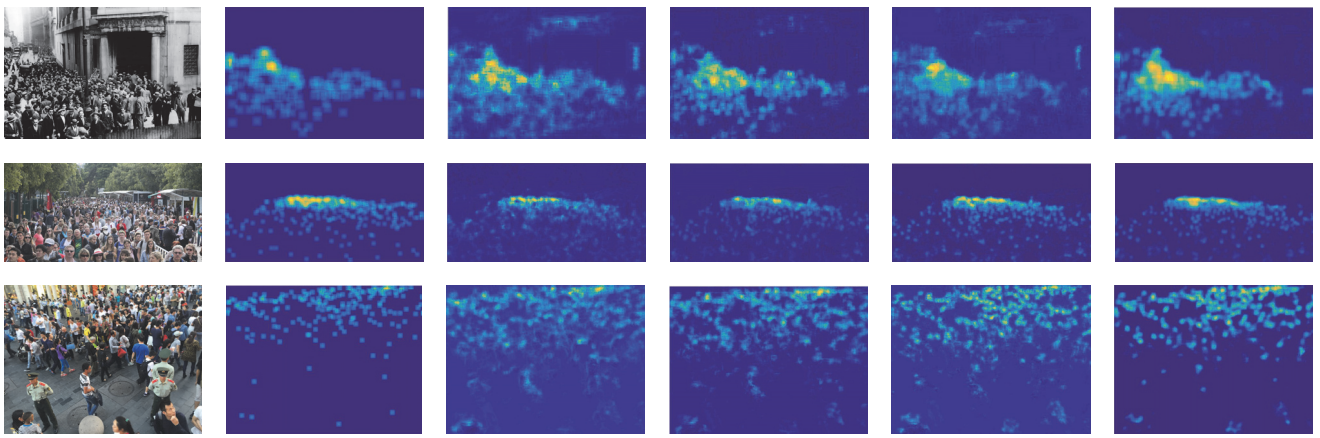


Figure 4: Density map regression performance of methods.

detailed view, we need to know their performance on pedestrian with diverse sizes. As the size of the pedestrian is always inversely proportional with the crowd density in extensively crowded images, we divide the two datasets to subgroups according to the number of pedestrians in the images. The performance of CNN based methods is presented in Figure 4.

The x-axis denotes the number of pedestrians in an image and with the increase of the number of pedestrians, the average size of each one is decreased. In Figure 5, when the number of pedestrians is under 50, which denotes that average size of pedestrian is large, the MRE decrease by about 50% compared with [11] and by about 35% compared with [12]. When the crowd count is above 350 which demonstrates that the pedestrians are small, the MRE is about half the MRE of [12] and only a third of the MRE of [11], which not only shows the robustness of our method, but shows our methods' strong capability on small instance capturing.

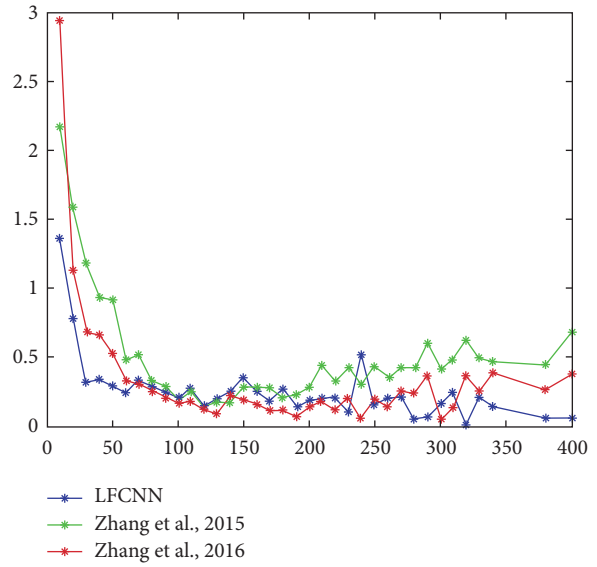


Figure 5: MRE of the CNN based methods in the WorldExpo10 dataset evaluated in different subgroups.

7. Conclusion

In this paper, we have proposed an accurate crowd counting method, called low-rank and sparse based deep-fusion convolutional neural network *(LFCNN)*. In this method, the proposed deep-fusion network is designed to capture the multiscale targets and promote the density map regression accuracy. Then the global counting is regressed through a low-rank and sparse based regression. To our knowledge, the low-rank and sparse penalty is firstly used for the regression of global counting. Experiments on large-scale crowd counting datasets demonstrate the promotion of accuracy achieved by the proposed method.

Disclosure

The paper matches the formatting instructions of IJCAI-07.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61473149). The support of IJCAI, Inc., is acknowledged.

References

[1] S. A. M. Saleh, S. A. Suandi, and H. Ibrahim, "Recent survey on crowd density estimation and counting for visual surveillance," *Engineering Applications of Artificial Intelligence*, vol. 41, pp. 103–114, 2015.

[2] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15)*, pp. 1–9, Boston, Mass, USA, June 2015.

[3] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," in *Proceedings of the 2nd*

IEEE/IAPR International Joint Conference on Biometrics, IJCB 2014, pp. 1–8, October 2014.

[4] Z. Ma, L. Yu, and A. B. Chan, "Small instance detection by integer programming on object density maps," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 3689–3697, June 2015.

[5] V. Rabaud and S. Belongie, "Counting crowded moving objects," in *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 705–711, June 2006.

[6] R. Ma, L. Li, W. Huang, and Q. Tian, "On pixel count based crowd density estimation for visual surveillance," in *Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems*, vol. 1, pp. 170–173, December 2004.

[7] H. Rahmalan, M. Nixon, and J. Carter, "On crowd density estimation for surveillance," in *Proceedings of the IET Conference on Crime and Security*, pp. 540–545, London, UK, 2006.

[8] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–7, June 2008.

[9] V. Lempitsky and A. Zisserman, "Learning to count objects in images," in *Proceedings of the 24th Annual Conference on Neural Information Processing Systems (NIPS '10)*, pp. 1324–1332, December 2010.

[10] L. Fiaschi, U. Koethe, R. Nair, and F. A. Hamprecht, "Learning to count with regression forest and structured labels," in *Proceedings of the 21st International Conference on Pattern Recognition, ICPR 2012*, pp. 2685–2688, November 2012.

[11] C. Zhang, H. Li, X. Wang, and X. Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR' 15)*, pp. 833–841, June 2015.

[12] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 589–597, July 2016.

[13] L. Boominathan, S. S. S. Kruthiventi, and R. Venkatesh Babu, "CrowdNet: A deep convolutional network for dense crowd counting," in *Proceedings of the 24th ACM Multimedia Conference, MM 2016*, pp. 640–644, October 2016.

[14] D. Onoro-Rubio and R. J. López-Sastre, "Towards perspective-free object counting with deep learning," in *Proceedings of the European Conference on Computer Vision*, pp. 615–629, Springer, 2016.

[15] B. Xu and G. Qiu, "Crowd density estimation based on rich features and random projection forest," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision, WACV 2016*, pp. 1–8, March 2016.

[16] K. Chen, C. C. Loy, S. Gong, and T. Xiang, "Feature mining for localised crowd counting," in *Proceedings of the 2012 23rd British Machine Vision Conference, BMVC 2012*, vol. 1, article 3, September 2012.

[17] V.-Q. Pham, T. Kozakaya, O. Yamaguchi, and R. Okada, "COUNT forest: Co-voting uncertain number of targets using random forest for crowd density estimation," in *Proceedings of the 15th IEEE International Conference on Computer Vision, ICCV 2015*, pp. 3253–3261, December 2015.

[18] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *Journal of Sensors*, vol. 2015, Article ID 258619, 12 pages, 2015.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 770–778, July 2016.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 779–788, July 2016.

[21] X. Lu, X. Duan, X. Mao, Y. Li, and X. Zhang, "Feature extraction and fusion using deep convolutional neural networks for face detection," *Mathematical Problems in Engineering*, vol. 2017, Article ID 1376726, 9 pages, 2017.

[22] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 580–587, Columbus, Ohio, USA, June 2014.

[23] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *Proceedings of the 14th IEEE International Conference on Computer Vision (ICCV '13)*, pp. 2056–2063, December 2013.

[24] E. Walach and L. Wolf, "Learning to count with CNN boosting," in *Proceedings of the European Conference on Computer Vision*, pp. 660–676, Springer.

[25] C. Wang, H. Zhang, L. Yang, S. Liu, and X. Cao, "Deep people counting in extremely dense crowds," in *Proceedings of the 23rd ACM International Conference on Multimedia, MM 2015*, pp. 1299–1302, October 2015.

[26] C. Shang, H. Ai, and B. Bai, "End-to-end crowd counting via joint learning local and global count," in *Proceedings of the 23rd IEEE International Conference on Image Processing, ICIP 2016*, pp. 1215–1219, September 2016.

[27] Y. Hu, H. Chang, F. Nian, Y. Wang, and T. Li, "Dense crowd counting from still images with convolutional neural networks," *Journal of Visual Communication and Image Representation*, vol. 38, pp. 530–539, 2016.

[28] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1–3, pp. 37–52, 1987.

[29] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, article 11, 2011.

[30] J. Wright, Y. Peng, Y. Ma, A. Ganesh, and S. Rao, "Robust principal component analysis: exact recovery of corrupted low-rank matrices by convex optimization," in *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems (NIPS '09)*, pp. 2080–2088, Vancouver, Canada, December 2009.

[31] T. Zhou and D. Tao, "GoDec: randomized low-rank & sparse matrix decomposition in noisy case," in *Proceedings of the 28th International Conference on Machine Learning (ICML '11)*, pp. 33–40, July 2011.

[32] R. Chalapathy, A. K. Menon, and S. Sanjay, *Robust, deep and inductive anomaly detection*, arXiv preprint https://arxiv.org/abs/1704.06743.

[33] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," *Journal of Machine Learning Research*, vol. 15, pp. 315–323, 2011.

[34] R. A. Horn, "The hadamard product," in *Proceedings of the Symposia in Applied Mathematics*, vol. 40, pp. 87–169, 1990.

[35] J. Chen, J. Zhou, and J. Ye, "Integrating low-rank and group-sparse structures for robust multi-task learning," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'11*, pp. 42–50, San Diego, Calif, USA, August 2011.

[36] Y. Jia, E. Shelhamer, J. Donahue et al., "Caffe: convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*, pp. 675–678, ACM, Orlando, Fla, USA, November 2014.

[37] J. Zhou, J. Chen, and J. Ye, *Malsar, Multi-Task Learning via Structural Regularization*, vol. 21, Arizona State University, 2011.

[38] A. B. Chan and N. Vasconcelos, "Counting people with low-level features and Bayesian regression," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2160–2177, 2012.

[39] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1975–1981, San Francisco, Calif, USA, June 2010.

[40] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[41] J. A. Suykens, T. Van Gestel, and J. De Brabanter, "Least squares support vector machines," *World Scientific*, 2002.

[42] T. Weldon and W. E. Higgins, "Designing multiple Gabor filters for multitexture image segmentation," *Optical Engineering*, vol. 38, no. 9, pp. 1478–1489, 1999.

[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, Lake Tahoe, Nev, USA, December 2012.

[44] Karen. S. and Z. Andrew, "Very deep convolutional networks for large-scale image recognition," 2014, arXiv preprint https://arxiv.org/abs/1409.1556.