Pedestrian Density Analysis in Public Scenes With Spatiotemporal Tensor Features

Ke Chen and Joni-Kristian Kämäräinen, *Member, IEEE*

***Abstract*—Pedestrian density estimation is one of the key problems in intelligent transportation systems and has been widely applied to a number of applications in other fields of engineering. Counting-by-regression methods are more favorable for coping with such a problem owing to their robustness against interperson occlusion and relaxing the impractical requirement of a high video frame rate, compared to counting-by-detection and counting-by-clustering methods. However, imagery features in the existing counting-by-regression approaches are extracted from the whole region or spatially localized cells/pixels of each single video frame, which omits the unique motion patterns of the same pedestrians across the neighboring frames. In the light of this, this paper exploits a novel tensor-formed spatiotemporal feature representation and applies it in a multilinear regression learning framework, which can capture spatially distributed dynamic crowd patterns by discovering the latent multidimensional structural correlations of tensor features along both spatial (i.e., horizontal and vertical) and temporal dimensions. Extensive evaluation with the public UCSD and Shopping Mall benchmarks demonstrate superior performance of our approach to the state-of-the-art counting methods even when the surveillance data has a low frame rate.**

***Index Terms*—Pedestrian density analysis, spatiotemporal features, tensor, regression, multilinear learning.**

I. Introduction

PEDESTRAIN density analysis under a monocular surveillance camera is one of the active research topics, which has its significance in designing and monitoring public transportation system [1]–[3]. Specifically, pedestrian density estimation is to predict the exact number of pedestrians in global or spatially-distributed locations of public scenes, which can also be termed as pedestrian counting problem. The crowd density can be a useful information in other related problems such as person detection and tracking [4], anomaly behavior detection [5], and dynamic crowd pattern analysis [6]. The problem remains challenging due to the low resolution of surveillance videos, the occlusion between people, and severe perspective distortion.

To cope with such a problem, a number of algorithms have been proposed, which can fall into three main categories: counting-by-detection [7]–[9], counting-by-clustering [10], [11] and counting-by-regression [2], [12]–[24]. Counting-by-detection frameworks [7]–[9] calculate the number of detected pedestrians and can accurately locate persons at the cost of time-consuming exhaustive scan of the whole image space and are less robust against inter-person occlusion. Counting-by-clustering methods [10], [11] assume that each pedestrian has a unique coherent motion pattern and work well only with a sufficiently high video frame rate to obtain reliable motion information. Counting-by-regression methods, [2], [12]–[23] learn a regression mapping from low-level imagery features to the pedestrian count directly and are able to overcome the aforementioned drawbacks. Specifically, counting-by-regression methods are computationally fast, more robust against occlusions as compared to counting-by-detection, and can relax the strong assumption of temporal motion tracking of counting-by-clustering frameworks.

Existing regression-based approaches [13], [15], [19] mainly focus on exploiting globally the robust and informative feature representation to learn a discriminative linear combination function. Some efforts were devoted to adding location information to feature space at the cell-level [14] or pixel-level [25], [26]. The aforementioned imagery representation was limited to the spatial domain, which inspired a number of works to discover and utilize the temporal correlation of features across video frames [27]–[31]. These spatiotemporal frameworks are aimed at estimating the number of pedestrian flow crossing the Line of Interest (LOI or termed as the virtual gate), which were constrained by the settings of virtual gates and dependent on a high frame rate (e.g., the frame rate of the datasets used in [27]–[31] is at least 7 fps [19]). In practice, the setting of a high frame rate increases the requirement of hardware, data transmission and computational complexity of algorithms in visual surveillance systems. Consequently, incorporating temporal information into the existing counting-by-regression frameworks and the capability of handing with a low frame rate data motivate us to develop novel spatiotemporal features using tensor representation, which is suitable for complex real-world applications because of the representation containing multi-dimensional dependency (i.e., the spatial and temporal correlation in pedestrian density estimation).

We consider that mining spatial and temporal correlations of low-level local features from adjacent cells between neighboring frames is important to achieve accurate and robust counting results in a unique regression framework, which is missing in all existing techniques. Intuitively, the concept of our method is extremely simple: crowd density pattern in each localized region seeks support from neighboring regions (cells) in a temporal

Manuscript received June 6, 2015; revised November 13, 2015; accepted December 28, 2015. Date of publication March 28, 2016; date of current version June 24, 2016. This work was supported in part by the Academy of Finland under Grant 267581, by the Digile Oy and Nokia Technologies (Tampere, Finland) through the D2I SHOK Project, and by the CSC-IT Center for Science, Finland. The Associate Editor for this paper was B. Morris.

The authors are with the Department of Signal Processing, Tampere University of Technology, 33720 Tampere, Finland (e-mail: ke.chen@tut.fi; joni.kamarainen@tut.fi).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2016.2516586

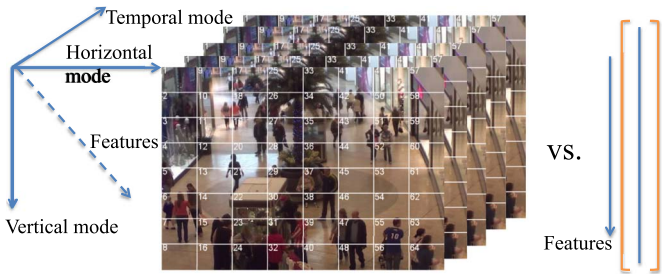


Fig. 1. Tensor-formed vs. vector-formed imagery representation. The latent structural information underlying the video can be mined by tensor decomposition along each mode, i.e., spatial (horizontal and vertical) and temporal dimensions. For exploiting the rich structure of tensor feature representation instead of vector-formed features, we adopt a novel tensor learning framework to capture dynamic crowd patterns in both spatial and temporal domains.

sliding window. To achieve that, an intermediate spatiotemporal descriptor for pedestrian density estimation is developed in the form of tensor, which is learned by multilinear regression models to mine the underlying latent structure (i.e., the latent spatial and temporal correlations of dynamic localized crowd patterns). The aim of our proposed framework can be achieved by tensor decomposition, which will be constrained to learn a linear combination of feature components at each mode (order) of tensor. In other words, the multilinear learning model with the tensor feature learns the regression weights by addressing multiple linear mapping problems constrained in each domain simultaneously. This is different from other frameworks that use a simple linear combination of the concatenated vectors. The difference between tensor-formed features and concatenated vector features adopted in the existing literatures for pedestrian counting are illustrated in Fig. 1. Moreover, the proposed spatiotemporal tensor representation for video frames can relax the strong requirement of a high frame rate and cope with bigger temporal difference between the frames by simply adjusting the size of temporal sliding window. The main contributions and novelties of this study are three-fold:

- An intermediate spatiotemporal feature representation based on localized regions is for the first time exploited in the form of tensor to incorporate both spatial and temporal information for capturing dynamic pedestrian patterns in the cases of spatial Region of Interest (ROI) and temporal Line of Interest (LOI) in pedestrian counting.
- A multilinear learning model, i.e., tensor ridge regression, is introduced for the first time to estimate pedestrian density by mining the latent structural information of spatiotemporal tensors along both spatial and temporal dimensions (tensor modes).
- Extensive experiments on two public benchmarks for crowd density estimation demonstrate the superior performance achieved by our method over the state-of-the-arts, even for the data with a low frame rate.

II. Related Work

Pedestrian Density Estimation: The existing algorithms for estimating the density of crowd in public scenes are either detection based [7]–[9], clustering based [10], [11] or regression based [2], [12]–[23]. Counting persons in the scenes in a regression framework is more suitable for public crowded scenes owing to its robustness against inter-person occlusion and is thus widely adopted in the recent work [2], [12]–[23]. In [12], [13], [15], [16], [19], imagery feature representation extracted from the entire spatial space were mapped to a scalar-valued continuous count label. However, global features missed the important location information that provides spatially varying crowd patterns caused by density, scene layout, and self-organization of crowd [32]. In light of this, localized features based on cells [14], [23] or pixels [25], [26] have been employed relaxing the global assumption and adapting spatially aware features which are more accurate and reliable for crowd counting. Nevertheless, the existing regression frameworks for crowd density estimation assumed implicitly that the models are linear combination of the spatial features, which is largely invalid in real-worlds scenarios. In particular, crowd density pattern can vary differently according to its location in spatial and temporal domains, which can not be described by a simple linear combination. For the purpose of learning the latent structural information underlying the crowd structures in spatial and temporal domains, we propose a spatiotemporal tensor feature representation, which can be learned effectively by a multilinear learning model.

Multilinear Learning for Tensor Data: Tensor encoded data with multiple dimensions have received increasing attention, which is widely encountered in science and engineering [33]–[39]. Tensors having rich structural information can be utilized in different ways according to the multi-dimensional dependency of the specific problems. Recently, Guo *et al.* [33] exploited a tensor-to-scalar learning framework and applied it to a number of applications in still images. Geng *et al.* [36] proposed an alternative way by discovering the sample correlation in pose, illumination, and person tensor modes, which can cope with the missing value problem. Zhao *et al.* [37] developed a tensor-to-tensor regressor to discover and utilize the complex correlation between multi-output tensor elements. In this paper, considering the characteristics of crowd density estimation problem, a multilinear model (i.e., tensor ridge regression) is employed to capture not only spatial but also complex temporal correlation of localized features. Note that, any existing multilinear regressor can be adopted here and we choose tensor ridge regression in this paper because of its simple structure and the significance as the direct competitor to ordinary ridge regression in the existing work [14]–[16]. Experiment results demonstrate the effectiveness and superior performance of exploiting the tensor-formed spatiotemporal features to the state-of-the-art methods in pedestrian density analysis.

Pipeline and Paper Structure: As shown in Fig. 2, an overview of our framework is given with the following steps:

- Given a set of training images, we extract low-level spatial imagery features from localized regions including segment-based features, edge-based features, and local texture features. A spatiotemporal tensor-formed feature can then be constructed to combine the spatial tensor features from a sliding window of frames in the temporal domain (Section III).

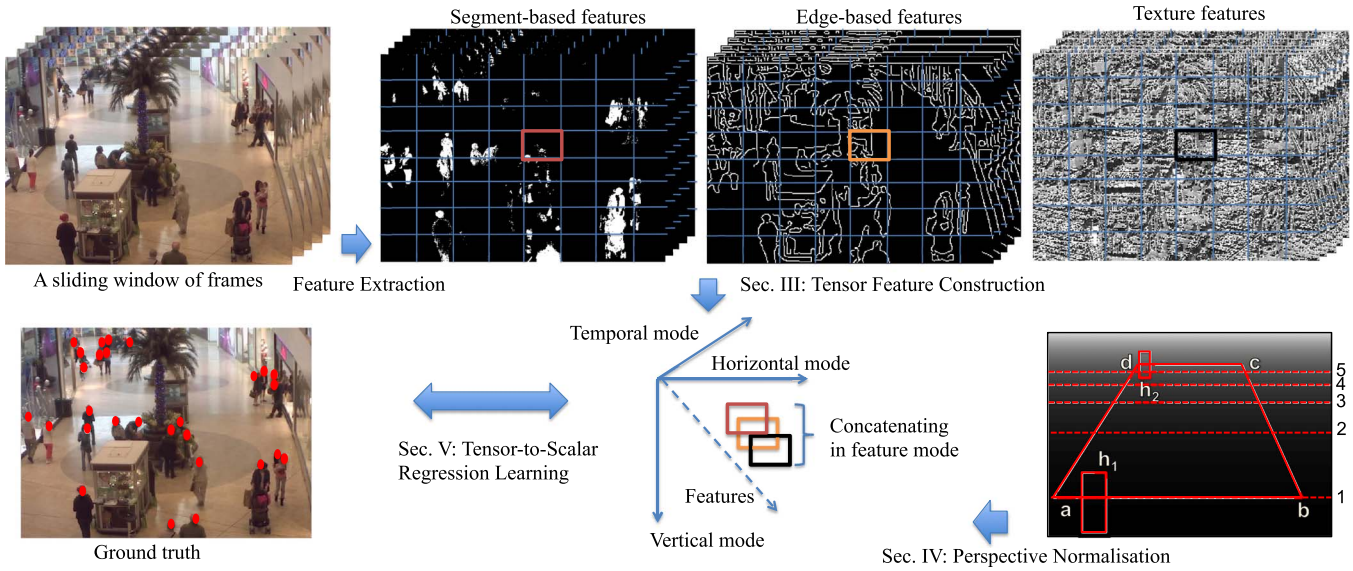


Fig. 2. The pipeline of a multilinear regression learning framework using the proposed spatiotemporal tensor feature representation.

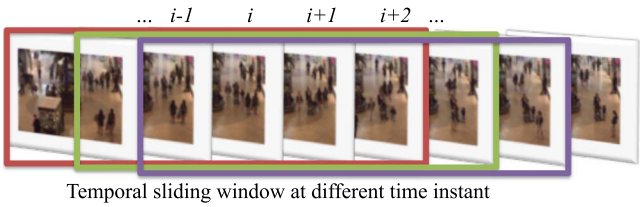


Fig. 3. A short temporal sliding window shot used by our framework.

- To alleviate the issues of perspective distortion, we infer a map which normalize the perspective effects by measuring the ratio of changes of an object at locations near and far away from the camera (Section IV).
- A multilinear regression model is trained using the spatiotemporal tensor feature inputs and the scalar-valued pedestrian count as a training pair (Section V). The model is applied to the two popular problems of pedestrian counting: region of interest and crossing virtual gates.

Given a new test frame, features are first extracted and then fed into the learned multilinear regression model for estimating a continuous value indicating the crowd count in the scene.

III. Spatiotemporal Tensor Features

We denote the current video frame by i, $i = 1, 2, \ldots, N$ where N denotes the total number of training frames and the corresponding ground truth pedestrian count is y_i. We first generate training video shots within a temporal sliding window consisted of the frames $F = \{i-k, i-k+1, \ldots, i, i+1, \ldots, i+k\}$ with the total of $\|F\| = 2k+1$ frames in each shot. The reference frame and its count value are $\langle i, y_i \rangle$ (Fig. 3). Consequently, we will have N video shots for training, where the first and last k shots have imbalanced window width to keep the consistent window size and the remaining shots are anchored with the center of the reference frame and a balanced window width. We then spatially divide each frame of the ith video shot into $H \times W$ grid regions to form a spatiotemporal volumes consisting of $H \times W \times F$ localized cells. For each localized cell, three types of features are extracted to form a D-dimensional feature vector ($D = 30$ in our case) as in [12], [14]–[16], [19], [20] with more details given in Table I as well as the following paragraphs. Before extracting the features, we transform the frames into gray-scale.

TABLE I
Contents of the Low Level Feature Vector

	Feature	*Dims*	*Params*
Segment	Area	1	–
	Perimeter	1	–
	Perimeter-area ratio	1	–
	Perimeter orientation	6	$0^o, 30^o, 60^o, 90^o, 120^o, 150^o$
	Blob count	1	the size of blob $>$ 10 pixel
Edge	Edge	1	–
	Edge orientation	6	$0^o, 30^o, 60^o, 90^o, 120^o, 150^o$
	Edge Minkowski	1	–
Texture	Texture homogeneity	4	$0^o, 45^o, 90^o, 135^o$
	Texture energy	4	$0^o, 45^o, 90^o, 135^o$
	Texture entropy	4	$0^o, 45^o, 90^o, 135^o$

Segment-Based Features: Foreground segmentation is a widely adopted technique for crowd density estimation because of its simple and direct correspondence between pedestrian count and the statistics of foreground segments. To obtain the foreground, background subtraction, such as mixture of Gaussians-based technique [40] or mixture of dynamic textures-based method [41], is utilized with generating the following features from the extracted foreground segment:

- Area—a total number of pixels in the segment.
- Perimeter—a total number of pixels on the segment perimeter.
- Perimeter-area ratio—a ratio between the segment perimeter and area to measure the complexity of the segment shape.
- Perimeter orientation—orientation histogram of the segment perimeter.
- Blob count—the number of connected components with area larger than a predefined threshold.

Edge-Based Features: Foreground segment based features can capture the global properties of moving segments, but severe inter-person occlusions often appearing in highly crowded scenes make them less reliable. This has motivated researchers to take edge based features into account to measure occlusion edges [12], [14]–[16], [18]–[20], [42]. In this work, we adopt the following features with the edges detected by Canny edge detector [43]:

- Edge—total number of edge pixels.
- Edge orientation—histogram of the edge orientations in the segment.
- Minkowski dimension—the Minkowski fractal dimension or box-counting dimension of the edges [44], which counts how many pre-defined structuring elements are required to fill the edges.

To reduce the negative effect of irrelevant edges, we apply the binary foreground segments as a mask to the Canny edge image prior to feature extraction.

Texture Features: Beyond the segment-based and edge-based features, the crowd texture and gradient patterns reflect the appearance and shape of local objects, e.g., the human head and shoulders, and carry additional cues about pedestrian density. Gray-Level Co-occurrence Matrix (GLCM) [45] is widely used in various crowd counting studies [12], [14]–[16], [19], [20], [23]. Typically, we quantize the image masked by the foreground segment into 8 gray-levels and then estimate the joint probability or co-occurrence of neighboring pixel values, $p(i, j \mid \theta)$ for four orientations, which are given in Table I. After extracting the co-occurrence matrix, a set of features can be derived for each θ.

- Texture homogeneity—a smoothness measure $g_\theta = \sum_{i,j}(p(i, j \mid \theta)/1 + |i - j|)$.
- Texture energy—total sum-squared energy $e_\theta = \sum_{i,j} p(i, j \mid \theta)^2$.
- Texture entropy—a randomness measure $h_\theta = \sum_{i,j} p(i, j \mid \theta) \log p(i, j \mid \theta)$.

Spatiotemporal Tensor Features: After the extraction of D-dimensional segment, edge and texture features described above, an intermediate tensor-formed spatiotemporal imagery feature is constructed from each cell region (h, w) and each frame f of ith training shot:

$$\mathcal{X}_i \in \mathbb{R}^{H \times W \times F \times D}. \tag{1}$$

Given the actual people count y_i in the reference frame of each shot, we obtain the training pairs $\{\mathcal{X}, y\}_i, i = 1, 2, \ldots, N$, which are used as input and output in supervised training of a multilinear regression model.

IV. Perspective Normalization

The problem of perspective distortion, i.e., the object appearing in a smaller size when moving away from the surveillance camera, can make the extracted features from different spatial locations in the scene inconsistent, which increases the difficulty in training regression models. The straightforward solution to mitigate the suffering of perspective distortion is to divide the whole images into a number of local regions, each of which is modeled by a separate regression function [23]. However, the price of using numerous local regressors is to increase the computational cost and not scalable. The alternative approach is perspective normalization which is performed to bring perceived size of objects at different depths to the same scale by multiplying each pixel by a weight (the farther away from the camera the pixels are, the larger weight they use) before the features are fed to train regression models.

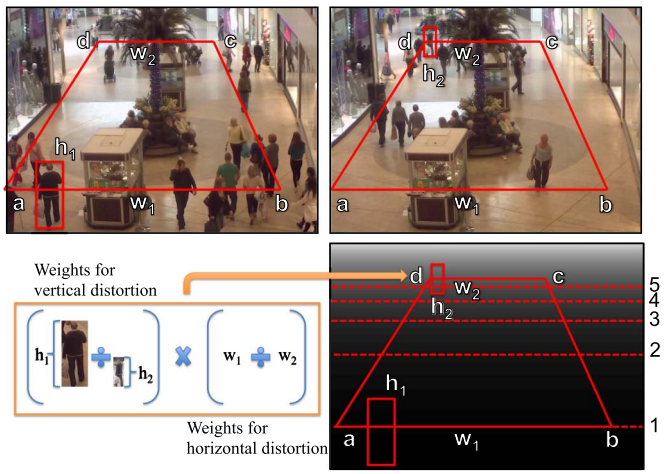


Fig. 4. A perspective normalization map can be generated by linear interpolation according to the relative changes of a reference person's height and the width of horizontal line of a predefined quadrilateral at different depth in the scene.

The typical method based on linear interpolation for perspective normalization widely adopted in recent work [12], [14]–[16], [19], [20], [25], [26] is described in [12], [19]. The work flow to generate a perspective map is illustrated in Fig. 4, which can be presented as the following steps:

1) Four landmarks (e.g., points a, b, c, and d in Fig. 4) in the scene is first selected to form a quadrilateral, which corresponds to a rectangle.
2) A reference pedestrian is selected to measure the vertical distortion on the horizontal lines $\overline{ab}$ and $\overline{cd}$ by using the ratio between the pedestrian's height h_1 and h_2 at two extremes, i.e., its bounding box's center touches the $\overline{ab}$ and $\overline{cd}$. Horizontal distortion can be easily employed the ratio of the width w_1 and w_2 of the $\overline{ab}$ and $\overline{cd}$. The weights at $\overline{ab}$ and $\overline{cd}$ are then assigned as 1 and $h_1 w_1 / h_2 w_2$ respectively.
3) Considering the pixel-level adjustment employed, the remaining weights of the scene can be determined using the corresponding linear interpolated h' and w' as well as the weight of the $\overline{ab}$ line and the height of the reference pedestrian passing $\overline{ab}$ (i.e., $h_1 w_1 / h' w'$, where the interpolants h' and w' can be computed by linear interpolation with using $\{h_1, w_1\}$ and $\{h_2, w_2\}$.

When applying the weights to features, the characteristics of different features must be taken into account [19]: for foreground segment pixels, the original weight of the perspective map is adopted; for the pixels of edge images, they are weighted by square-roots of the weights; and GLCM features are normalized by weighting the occurrence of each pixel pair when accumulating the co-occurrence matrix.

V. Regression Learning With Tensor Data

With the low-level imagery feature representation and pedestrian count $\{\mathcal{X}, y\}_i, i = 1, 2 \dots N$, a multilinear regression can be adopted to capture the latent correlation underlying in the feature tensors. Ridge regression has been proven its effectiveness in addressing crowd counting problem [14]–[16], [20], [46], which is also investigated in Section V-A for the completeness. The details of tensor ridge regression is presented in Section V-B.

A. Ridge Regression

Before giving more details about tensor ridge regression, we first introduce ridge regression given vector-formed features $\boldsymbol{x}_i \in \mathbb{R}^m$ and scalar-valued label $y_i \in \mathbb{R}$, which can be formulated as the following

$$\min \ \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{\mathrm{N}} \mathrm{loss}\left(y_i, f(\boldsymbol{x}_i; \boldsymbol{w}, b)\right) \tag{2}$$

where $\boldsymbol{w} \in \mathbb{R}^m$ is the weight vector to be optimized, $b \in \mathbb{R}$ is the bias term, and $C \in \mathbb{R}$ denotes the trade-off parameter between the regularized term and the loss function $\mathrm{loss}(\cdot)$. The regression mapping function is given as follows:

$$f(\boldsymbol{x}_i; \boldsymbol{w}, b) = \sum_{j=1}^{m} \boldsymbol{w}_j \boldsymbol{x}_{ij} + b = \boldsymbol{x}_i^T \boldsymbol{w} + b \tag{3}$$

where $\boldsymbol{w}_j$ and $\boldsymbol{x}_{ij}$ are jth element of vector $\boldsymbol{w}$ and $\boldsymbol{x}_i$ respectively.

With using a quadratic loss function $\mathrm{loss}(\boldsymbol{a}) = \|\boldsymbol{a}\|_2^2 = \boldsymbol{a}^T\boldsymbol{a}$, Equation (2) can be written as:

$$\min \ \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{\mathrm{N}} \|\langle \boldsymbol{x}_i, \boldsymbol{w}\rangle + b - y_i\|_2^2 . \tag{4}$$

Considering the differentiability of Equation (4), the gradient of such an object function is enforced to be zero, which thus has the following close-form solution:

$$\begin{bmatrix} \boldsymbol{w} \\ b \end{bmatrix} = -(\boldsymbol{Q}^T\boldsymbol{Q})^{-1}\boldsymbol{Q}^T\boldsymbol{p} \tag{5}$$

where positive semi-definite matrix $\boldsymbol{Q}$ and vector $\boldsymbol{p}$ can be obtained as

$$\begin{aligned} \boldsymbol{Q} &= \begin{bmatrix} 2C\sum_{i=1}^{\mathrm{N}} \boldsymbol{x}_i\boldsymbol{x}_i^T + \boldsymbol{I} & 2C\sum_{i=1}^{\mathrm{N}} \boldsymbol{x}_i \\ 2C\sum_{i=1}^{\mathrm{N}} \boldsymbol{x}_i^T & 2C\mathrm{N} \end{bmatrix} \\ \boldsymbol{p} &= \begin{bmatrix} -2C\sum_{i=1}^{\mathrm{N}} \boldsymbol{x}_i y_i^T \\ -2C\sum_{i=1}^{\mathrm{N}} y_i^T \end{bmatrix}. \end{aligned} \tag{6}$$

The parameter C is typically determined by n-fold cross validation.

B. Tensor Ridge Regression

Conventionally, ridge regression widely adopted in pedestrian density estimation [14]–[16], [20], [23] is to learn a discriminative linear function with a vector-to-scalar mapping. Recently, a multilinear extension based on ridge regression were proposed to discover the multi-dimensional latent dependency in tensor features [33], [35] with pairs of tensor input and scalar output. In addition to fair comparison with state-of-the-art frameworks based on ridge regression [14]–[16], [20], [23], tensor ridge regression (TRR) is employed to map the proposed spatiotemporal tensor feature representation to person count in this paper owing to its simple implementation and high computational efficiency.

In mathematics, given $\{\mathcal{X}, y\}_i, i = 1, 2 \dots N$ as the observation tensor and target scalar (person count), tensor ridge regression is aimed to learn a discriminative function that fits the tensor input and scalar output as the following:

$$\begin{aligned} f(\mathcal{X}; \mathcal{W}, b) &= \langle \mathcal{X}, \mathcal{W}\rangle + b \\ &= \sum_{h=1}^{H}\sum_{w=1}^{W}\sum_{f=1}^{F}\sum_{d=1}^{D} \mathcal{X}^{(h,w,f,d)}\mathcal{W}^{(h,w,f,d)} + b \end{aligned} \tag{7}$$

where $\mathcal{W} \in \mathbb{R}^{H\times W\times F\times D}$ is the weight tensor and $b \in \mathbb{R}$ is the bias value. By following the principle of CP composition to capture the latent structure of tensor data along each mode (intuitively discover the spatial and temporal correlation of feature representation), $\mathcal{W} \in \mathbb{R}^{H\times W\times F\times D}$ can be constrained to be a sum of $R := \mathrm{rank}(\mathcal{X})$ rank-1 tensors [47]:

$$\mathcal{W} \approx \sum_{r=1}^{\mathrm{R}} z_r^{(1)} \circ z_r^{(2)} \circ z_r^{(3)} \circ z_r^{(4)} \triangleq \left\{ Z^{(1)}, Z^{(2)}, Z^{(3)}, Z^{(4)} \right\} \tag{8}$$

where $\circ$ is the outer product of vectors and $Z^{(j)} := [z_1^{(j)}, z_2^{(j)}, \dots, z_R^{(j)}], j = 1, 2, 3, 4$. To learn the discriminative function $f(\mathcal{X}; \mathcal{W}, b)$, tensor ridge regression is to minimize the regularized risk:

$$\min \ \varepsilon = \frac{1}{2}\|\mathcal{W}\|_F^2 + C\sum_{i=1}^{\mathrm{N}} \|\langle \mathcal{X}_i, \mathcal{W}\rangle + b - y_i\|_2^2 \tag{9}$$

where $\|\cdot\|_F$ denotes the Frobenius-norm and C is a parameter that controls the trade-off between the loss function and regularized term.

In the multilinear learning framework, at each iteration, for solving $\mathcal{Z}^{(j)}$ parameters, keeping $\mathcal{Z}^{(k)}$, $k \neq j$ fixed, object function (9) is thus reduced after CP decomposition to a subproblem as the following:

$$\begin{aligned} \min \ \varepsilon_j &= \frac{1}{2}\mathrm{trace}\left(\mathcal{Z}^{(j)} \left(\mathcal{Z}^{(\hat{j})}\right)^T \mathcal{Z}^{(\hat{j})} \left(\mathcal{Z}^{(j)}\right)^T \right) \\ &\quad + C\sum_{i=1}^{\mathrm{N}} \left(\mathrm{trace}\left(\mathcal{Z}^{(j)} \left(\mathcal{Z}^{(\hat{j})}\right)^T \mathcal{X}_{i(j)}^T \right) + b - y_i \right)^2 \end{aligned} \tag{10}$$

where $\mathcal{Z}^{(\hat{j})}$ denotes the Khatri–Rao product of $\mathcal{Z}^{(k)}$, $k \neq j$ (e.g., when $j = 3$, $\mathcal{Z}^{(\hat{j})}$ is equal to the Khatri–Rao product of $\mathcal{Z}^{(1)}$, $\mathcal{Z}^{(2)}$, and $\mathcal{Z}^{(4)}$)} and $\mathcal{X}_{i(j)}$ is the j-th mode feature for solving $\mathcal{Z}^{(j)}$ parameters [33]. Similar to vector-formed ridge regression [48], for the optimal solution of the minimization (10), the following equations can thus be obtained along the gradient-descent direction [49], [50]:

$$\begin{aligned} \frac{\partial \varepsilon_j}{\partial \mathcal{Z}^{(j)}} &= -2C\sum_{i=1}^{\mathrm{N}} \left(\mathrm{trace}\left(\mathcal{Z}^{(j)} \left(\mathcal{Z}^{(\hat{j})}\right)^T \mathcal{X}_{i(j)}^T \right) + b - y_i \right) \\ &\quad \times \mathcal{X}_{i(j)}\mathcal{Z}^{(\hat{j})} + \mathcal{Z}^{(j)} \left(\mathcal{Z}^{(\hat{j})}\right)^T \mathcal{Z}^{(\hat{j})} = 0 \\ \frac{\partial \varepsilon_j}{\partial b} &= -2C\sum_{i=1}^{\mathrm{N}} \left(\mathrm{trace}\left(\mathcal{Z}^{(j)} \left(\mathcal{Z}^{(\hat{j})}\right)^T \mathcal{X}_{i(j)}^T \right) + b - y_i \right) = 0. \end{aligned}$$

Evidently, the bias term b has the following closed-form solution dependent on unknown $\mathcal{Z}^{(j)}$ as

$$b = \frac{1}{N}\sum_{i=1}^{N}\left(\text{trace}\left(\mathcal{Z}^{(j)}\left(\mathcal{Z}^{(\hat{j})}\right)^{T}\mathcal{X}_{i(j)}^{T}\right) - y_i\right) \tag{11}$$

whereas we cannot obtain the closed-form solution for $\mathcal{Z}^{(j)}$ due to the failure of vectorization in view of the existence of $(\mathcal{Z}^{(\hat{j})})^{T}\mathcal{Z}^{(\hat{j})}$. Similar to [33], the regularized term (i.e., the former) in (10) can be transformed into the sum of separable regularization in each mode as [33] to achieve a close-form solution:

$$\zeta := \frac{1}{2}\text{trace}\left(\mathcal{Z}^{(j)}\left(\mathcal{Z}^{(\hat{j})}\right)^{T}\mathcal{Z}^{(\hat{j})}\left(\mathcal{Z}^{(j)}\right)^{T}\right) = \frac{1}{2}\sum_{l=1}^{4}\left\|\mathcal{Z}^{(l)}\right\|_{F}^{2}. \tag{12}$$

As a result, the partial derivative of the separable regularization term (12) with respect to $\mathcal{Z}^{(j)}$ can be written as

$$\frac{\partial\zeta}{\partial\mathcal{Z}^{(j)}} = \mathcal{Z}^{(j)}. \tag{13}$$

Evidently, $\|\mathcal{Z}^{(j)}\|_F^2 = \|\text{vec}(\mathcal{Z}^{(j)})\|_2^2$, where $\text{vec}(\mathcal{Z}^{(j)})$ generates a column vector obtained by stacking each column vectors of $\mathcal{Z}^{(j)}$ together [50]. It is worth mentioning here that ζ adopt an independent assumption to relax the correlation across each mode. However, owing to the existence of $(\mathcal{Z}^{(\hat{j})})^{T}\mathcal{Z}^{(\hat{j})}$ in the latter term of Equation (10), tensor ridge regression model can still benefit from mining rich structure of tensor-formed input $\mathcal{X}$. In view of $\text{trace}(\mathcal{Z}^{(j)}(\mathcal{Z}^{(\hat{j})})^{T}\mathcal{X}_{i(j)}^{T}) = \text{vec}(\mathcal{Z}^{(j)})^{T}\text{vec}(\mathcal{X}_{i(j)}\mathcal{Z}^{(\hat{j})})$, the closed form solution can be readily derived via matrix inversion as the following:

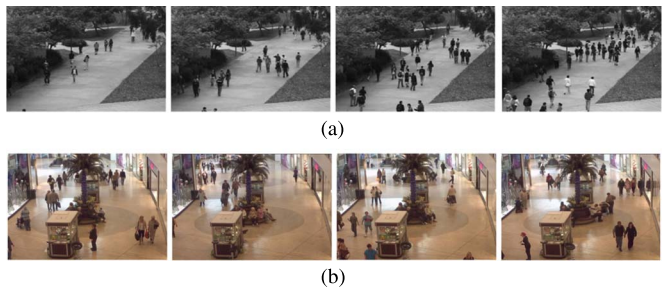
$$\mathbf{z}^{(j)} = (2C\boldsymbol{K}^{T}\boldsymbol{K} + \boldsymbol{I})^{-1}\boldsymbol{K}^{T}\mathbf{y} \tag{14}$$

where the unknown parameter vector $\boldsymbol{z}^{(j)}$ generated by stacking all the elements of $\mathcal{Z}^{(j)}$ and bias term b to be optimized, $\boldsymbol{K} \in \mathbb{R}^{N\times(N+1)}$ consisted of $\boldsymbol{k} := [\text{vec}(\mathcal{X}_{i(j)}\mathcal{Z}^{(\hat{j})})^{T}, 1] \in \mathbb{R}^{1\times(N+1)}$, $i = 1, 2 \ldots N$, $\boldsymbol{I} \in \mathbb{R}^{(N+1)\times(N+1)}$ is the identity matrix, and $\mathbf{y} := [y_1, y_2, \ldots, y_N]^{T} \in \mathbb{R}^{N}$.

Note that, the tensor-to-scalar mapping in multilinear learning is along different mode each iteration until the model converges, which indicates that the model will optimize the parameters in multiple regression learning directions to capture the latent structural information represented by tensor features. Intuitively, using the proposed spatiotemporal tensor features for analyzing pedestrian density, the dynamic crowd pattern across spatial regions and temporal frames can be discovered. The rational for exploiting Tensor Ridge Regression is that the model can cope with multicolinearity problem owing to its regularized least-square error minimization with superior robustness than ordinary least square regression methods [14], [51] and also it has a simple closed-form solution suitable for practical applications such as pedestrian density estimation. Other multilinear regression methods such as Support Tensor Regression (STR) [33] and High-Order Partial Least Square Regression (HOPLS) [37] can also be employed.

TABLE II
DATASET DETAILS: N_f = NUMBER OF VIDEO FRAMES, FPS = FRAME PER SECOND, R = RANGE OF OBJECT COUNT, AND Tp = TOTAL NUMBER OF PEDESTRIAN INSTANCES

Data	N_f	*FPS*	R	Tp
UCSD [12]	2000	10	11–46	49885
Mall [14]	2000	<2	13–53	62325



(a)

(b)

Fig. 5. The two popular benchmark datasets used in our experiments. (a) UCSD; (b) mall.

VI. EXPERIMENTS

To validate the effectiveness and generality of our multilinear counting-by-regression framework with spatiotemporal tensor features, the experiments were conducted to estimate the density of pedestrians within spatial Region of Interest (ROI) and crossing temporal Line of Interest (LOI), respectively. Finally, the discussion about multi-dimensional correlation (along spatial and temporal mode) of tensor-formed features was presented to visualize and verify the rationale of the proposed framework.

A. Datasets and Settings

Datasets: Experiments were conducted on the popular UCSD [12]–[16] and the Mall [14]–[16] benchmarks which feature realistic outdoor and indoor public scenes, respectively. Technical details of the datasets are presented in Table II and the illustrative examples of low to high pedestrian density are shown in Fig. 5. The Mall dataset is more challenging compared to the UCSD dataset in the light of the following reasons:

- lower frame rate leading to increased difficulty in utilizing temporal information;
- more dense crowd density causing more frequent inter-person occlusions as well as the occlusion by the scene objects;
- the changes of lighting conditions and glass surface reflections making feature representation inconsistent; and
- different activity patterns in the scene (both still and moving pedestrians).

Settings: For the UCSD dataset, we followed the same training and testing partition as in [12]–[16], i.e., Frames 601–1400 were used for training and the remaining frames for testing. For the Mall dataset, we used the first 800 frames for training and kept the rest 1200 frames for testing as [14]–[16]. For generating spatial regions, we follow the same cell-size setting in [14]: 6×4-cells for the UCSD dataset and 8×8-cells for the Mall dataset. In time domain, the size of a temporal sliding window of frames ($\|F\| = 5$ for both datasets) is selected by cross-validation.

TABLE III
COMPARATIVE EVALUATION WITH STATE-OF-THE-ARTS

Method	UCSD [12]			Mall [14]		
	mae	mse	mde	mae	mse	mde
MLR [23]	2.60	10.1	0.124	3.90	23.9	0.119
GPR [12]	2.30	8.21	0.114	3.72	20.1	0.115
RR [14]	2.25	7.82	0.110	3.59	19.0	0.110
MORR [14]	2.29	8.08	0.108	3.15	15.7	0.098
WRR [16]	2.11	7.11	0.105	3.58	19.0	0.110
CA-RR [15]	2.07	6.86	0.102	3.43	17.7	0.105
TRR	**1.97**	**5.86**	**0.091**	**2.59**	**11.0**	**0.081**

Different widths of the sliding window were evaluated and illustrated its effect on the performance in the experiments.

Evaluation Metrics: We employed three common metrics:

- mean absolute error: mae $= (1/N)\sum_{i=1}^{N}|v_i - \hat{v}_i|$,
- mean squared error: mse $= (1/N)\sum_{i=1}^{N}(v_i - \hat{v}_i)^2$, and
- mean deviation error: mde $= (1/N)\sum_{i=1}^{N}(|v_i - \hat{v}_i|/v_i)$,

where v_i is the actual count and $\hat{v}_i$ is the estimated count.

Comparative Methods: We compared our results to the following recent methods:

- Multiple Localized Regressors (MLR) [23]. The input and output for the model is the feature within each cell and the people count in the corresponding cell respectively. The ridge regression model is used to eliminate the effect of using different regression models.
- Single global model with global feature (1) ridge regression (RR) [52], and (2) Gaussian processes regression (GPR) with linear + RBF kernel as in [12]. These models employ global features as their input and the crowd density of the whole image as their output.
- Multi-output ridge regression (MORR) model with linear kernel [14]. The features and pedestrian count from each localized cell are concatenated into a vector-formed feature input and label output respectively.
- Weighted ridge regression (WRR) [16]. Weighted ridge regression with linear kernel with global features as input, pedestrian density as output and back-propagated error as sample weights.
- Ridge regression with cumulative attributes (CA-RR) [15]. A cumulative attribute space is inserted into low-level global features and global pedestrian count, which utilizes ridge regression with linear kernel as the second layer regression model.
- The proposed framework with spatiotemporal tensor features and tensor ridge regression (TRR) described in Section V-B.

For all models, their free parameters were tuned using 4-fold cross-validation.

B. Comparison With State-of-the-Arts

Performance Evaluation on Region of Interest: Table III compares crowd estimation performances of the seven regression-based frameworks and using the both benchmarking datasets. The results show that our tensor ridge regression (TRR) with spatiotemporal tensor features can achieve the best performance for both datasets and using all three metrics in comparison with state-of-the-art methods. The direct competitors using the variants of ridge regression with linear kernel are MLR [23], RR [14], MORR [14], WRR [16], and CA-RR [15]. Compared to MORR [14] with considering spatial correlation by multi-output ridge regression, our TRR can beat it owing to mining additional dependency in the temporal domain. Recently, state-of-the-art cumulative attribute learning framework with ridge regression (CA-RR) for pedestrian density estimation was proposed by Chen *et al.* [15] by introducing an intermediate attribute representation. However, TRR can still significantly outperform CA-RR by reducing at least 4.83% error on the UCSD benchmark and 24.4% error on the Mall dataset. It is worth pointing out that TRR can also incorporate the cumulative attributes to further improve the performance, which is out of the scope of this paper.

TABLE IV
PERFORMANCE EVALUATION WHEN CROSSING A VIRTUAL GATE (LINE OF INTEREST), WHERE REGIONS 1 AND 2 IN THE MALL DATASET CONSISTS OF CELLS 11, 12, 19, 20 AND CELLS 43, 44, 51, 52, RESPECTIVELY [14]

Method	Region 1 (R1)			Region 2 (R2)		
	mae	mse	mde	mae	mse	mde
MLR [23]	0.82	1.45	0.361	0.71	1.24	0.331
MORR [14]	0.76	1.22	0.331	**0.67**	1.12	0.306
TRR	**0.72**	**0.93**	**0.324**	**0.67**	**0.92**	**0.303**

Performance Evaluation on Crossing Virtual Gates: Table IV illustrates that superior performance can be achieved by our TRR model compared to the recent methods in [14], [23]. The explanation is that features from the corresponding regions (R1 and R2) and other nearby spatially-localized cells (context) in a video shot can provide additional information for the multilinear regressor, e.g., people likely to soon enter the virtual gate region. MORR method in [14] also considered spatial correlation across localized cells, but the simultaneous correlation discovery in the time domain contributes to our TRR to perform better. Note that the proposed model does not need to be altered for the virtual gate task, but it only affects to the regression outputs.

C. Evaluation on Tensor Features Construction

Tensor vs. Concatenated Vector Features: A key novelty of our model is the exploitation and usage of tensor feature representation. As explained in Section III, compared with ridge regression using the conventional concatenated vector formed features (RR-ST), the advantages of our tensor feature is that the latent structure of tensors can be learned by a multilinear regression model, i.e., tensor ridge regression. It is evident

TABLE V
COMPARISON WITH SPATIOTEMPORAL TENSOR AND CONCATENATED VECTOR FEATURES

Method	UCSD [12]			Mall [14]		
	mae	mse	mde	mae	mse	mde
RR-ST	2.21	7.48	0.105	2.78	12.3	0.086
TRR-ST	**1.97**	**5.86**	**0.091**	**2.59**	**11.0**	**0.081**

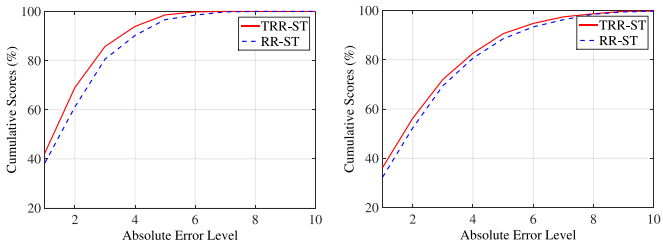


Fig. 6. Cumulative scores [15] with different values of error tolerance (the higher the better). Left: UCSD; Right: Mall.

TABLE VI
SPATIOTEMPORAL (TRR-ST) VS. SPATIAL TENSOR FEATURES (TRR-S)

Method	UCSD [12]			Mall [14]		
	mae	mse	mde	mae	mse	mde
TRR-S	2.26	7.83	0.106	3.18	15.6	0.097
TRR-ST	**1.97**	**5.86**	**0.091**	**2.59**	**11.0**	**0.081**

TABLE VII
SPATIOTEMPORAL (TRR-ST) VS. TEMPORAL TENSOR FEATURES (TRR-T)

Method	UCSD [12]			Mall [14]		
	mae	mse	mde	mae	mse	mde
TRR-T	2.20	7.54	0.107	3.39	17.8	0.104
TRR-ST	**1.97**	**5.86**	**0.091**	**2.59**	**11.0**	**0.081**

from Table V and Fig. 6 that constructing such tensor-formed features is a significant advantage for a regression framework that performs crowd counting.

Spatiotemporal vs. Spatial Tensor: Spatiotemporal and spatial tensor features are fed to the identical tensor ridge regression with linear kernel for counting pedestrians. The results illustrated in Table VI demonstrate that the latent correlated features in the neighboring frames can support to improve the performance of pedestrian density estimation.

Spatial-Temporal vs. Temporal Tensor: When using spatiotemporal features and temporal features of the global region in a number of frames for tensor ridge regression, the results in Table VII show that only considering the latent dependency in the temporal domain can be less effective due to omitting varying spatially-localized pedestrian density patterns compared to the proposed spatiotemporal tensor features.

Evaluation on the Size of Temporal Sliding Window: To show the size of temporal sliding window affecting on the performance, in Fig. 7, the number of frames in the sliding window is changed and the results are evaluated for the both datasets. We can observe that the performance of the UCSD dataset can be comparable especially when the window size is long enough ($\geqslant$5 frames, whereas the Mall dataset has a clear optimized size (i.e., 5 frames). It can be explained by the different frame rates

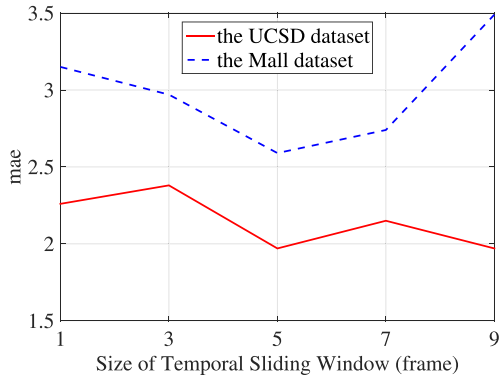


Fig. 7. Evaluation on the effect of the size of the temporal sliding window, i.e., the number of neighboring frames in the shot that forms the spatiotemporal tensor features.

TABLE VIII
EVALUATION ON THE EFFECT OF THE CELL-SIZE ON THE MALL BENCHMARK USING TRR MODEL

Cell-Size	Mall [14]		
	mae	mse	mde
1×1	3.39	17.8	0.104
2×2	2.89	12.8	0.090
4×4	2.79	12.3	0.086
8×8	2.59	11.0	0.081
16×16	2.62	11.1	0.082

of the two datasets. For example, for a video shot of 5 frames in the experiments, the time duration for the UCSD data is 0.5 seconds, while that for the Mall data it is more than 2 seconds. As a result, the UCSD dataset with much higher frame rate produces more correlated features between neighboring frames in the time domain as compared to the Mall dataset under the same settings, which makes regression learning for the Mall dataset more difficult. However, even with a very low frame rate, the proposed framework can also achieve significant improvement over the existing counting-by-regression methods.

Evaluation on the Spatial Cell Size: For the completeness of our work, the experiment to investigate the effect of the spatial cell-size was conducted. The results illustrated in Table VIII show that the counting performance improves with increasing the size of spatial cells until the optimal size is achieved, which supports our motivation about mining spatial correlation of features.

VII. CONCLUSION

This paper proposed a novel multilinear counting-by-regression framework with spatiotemporal tensor features to analyze the density of pedestrians in public scenes. Owing to the rich feature structure, the proposed framework can significantly outperform the state-of-the-art methods for crowd counting in the both practical settings: spatial Region of Interest and temporal Line of Interest, verified by the public USCD and Mall datasets. In our future work, we will address the two intriguing research questions, model transfer between datasets and automatic perspective correction, that will provide off-the-shelf solution with minimal involvement of humans for visual crowd density analysis.

REFERENCES

[1] G. Hamza-Lup, K. Hua, M. Le, and R. Peng, "Dynamic plan generation and real-time management techniques for traffic evacuation," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 4, pp. 615–624, Dec. 2008.

[2] J. Zhang, B. Tan, F. Sha, and L. He, "Predicting pedestrian counts in crowded scenes with rich and high-dimensional features," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1037–1046, Dec. 2011.

[3] A. Albiol, I. Mora, and V. Naranjo, "Real-time high density people counter using morphological tools," *IEEE Trans. Intell. Transp. Syst.*, vol. 2, no. 4, pp. 204–218, Dec. 2001.

[4] M. Rodriguez, I. Laptev, J. Sivic, and J.-Y. Audibert, "Density-aware person detection and tracking in crowds," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2423–2430.

[5] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 1446–1453.

[6] J. Shao, C. Loy, and X. Wang, "Scene-independent group profiling in crowd," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 2227–2234.

[7] W. Ge and R. Collins, "Marked point processes for crowd counting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 2913–2920.

[8] M. Li, Z. Zhang, K. Huang, and T. Tan, "Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection," in *Proc. Int. Conf. Pattern Recog.*, 2008, pp. 1–4.

[9] T. Zhao, R. Nevatia, and B. Wu, "Segmentation and tracking of multiple humans in crowded environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1198–1211, Jul. 2008.

[10] G. J. Brostow and R. Cipolla, "Unsupervised Bayesian detection of independent motion in crowds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 594–601.

[11] V. Rabaud and S. Belongie, "Counting crowded moving objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 705–711.

[12] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–7.

[13] A. B. Chan and N. Vasconcelos, "Counting people with low-level features and bayesian regression," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2160–2177, Apr. 2012.

[14] K. Chen, C. C. Loy, S. Gong, and T. Xiang, "Feature mining for localised crowd counting," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 21.1–21.11.

[15] K. Chen, S. Gong, T. Xiang, and C. C. Loy, "Cumulative attribute space for age and crowd density estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2467–2474.

[16] K. Chen and J.-K. Kämäräinen, "Learning to count with back-propagated information," in *Proc. Int. Conf. Pattern Recog.*, 2014, pp. 4672–4677.

[17] S. Cho, T. Chow, and C. Leung, "A neural-based crowd estimation by hybrid global learning algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 4, pp. 535–541, Aug. 1999.

[18] D. Kong, D. Gray, and H. Tao, "Counting pedestrians in crowds using viewpoint invariant training," presented at the British Machine Vision Conference, 2005.

[19] C. C. Loy, K. Chen, S. Gong, and T. Xiang, "Crowd counting and profiling: Methodology and evaluation," in *Proc. Model., Simul. Vis. Anal. Crowds*, 2013, pp. 347–382.

[20] C. C. Loy, S. Gong, and T. Xiang, "From semi-supervised to transfer counting of crowds," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2256–2263.

[21] A. Marana, S. Velastin, L. Costa, and R. Lotufo, "Estimation of crowd density using image processing," in *Proc. Image Process. Security Appl.*, 1997, pp. 11/1–11/8.

[22] D. Ryan, S. Denman, C. Fookes, and S. Sridharan, "Crowd counting using multiple local features," in *Proc. Digital Image Comput., Techn. Appl.*, 2009, pp. 81–88.

[23] X. Wu, G. Liang, K. Lee, and Y. Xu, "Crowd density estimation using texture analysis and learning," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2006, pp. 214–219.

[24] N. Tang, Y.-Y. Lin, M.-F. Weng, and H.-Y. Liao, "Cross-camera knowledge transfer for multiview people counting," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 80–93, Jan. 2015.

[25] V. Lempitsky and A. Zisserman, "Learning to count objects in images," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1324–1332.

[26] L. Fiaschi, R. Nair, U. Koethe, and F. A. Hamprecht, "Learning to count with regression forest and structured labels," in *Proc. Int. Conf. Pattern Recog.*, 2012, pp. 2685–2688.

[27] H. Yang, H. Su, S. Zheng, S. Wei, and Y. Fan, "The large-scale crowd density estimation based on sparse spatiotemporal local binary pattern," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2011, pp. 1–6.

[28] Y. Benabbas, N. Ihaddadene, T. Yahiaoui, T. Urruty, and C. Djeraba, "Spatio-temporal optical flow analysis for people counting," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveillance*, 2010, pp. 212–217.

[29] Z. Ma and A. B. Chan, "Crossing the line: Crowd counting by integer programming with local features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2539–2546.

[30] R. Liang, Y. Zhu, and H. Wang, "Counting crowd flow based on feature points," *Neurocomputing*, vol. 133, pp. 377–384, 2014.

[31] J. Wang, W. Fu, J. Liu, and H. Lu, "Spatiotemporal group context for pedestrian counting," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1620–1630, Sep. 2014.

[32] D. Helbing, A. Johansson, and E. T. Hochschule, "Pedestrian, crowd and evacuation dynamics," in *Encyclopedia of Complexity and Systems Science*. New York, NY, USA: Springer-Verlag, 2009.

[33] W. Guo, I. Kotsia, and I. Patras, "Tensor learning for regression," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 816–827, Feb. 2012.

[34] F. A. Andaló, P. A. V. Miranda, R. da S. Torres, and A. X. Falcão, "Shape feature extraction and description based on tensor scale," *Pattern Recog.*, vol. 43, no. 1, pp. 26–36, Jan. 2010.

[35] X. Wu and J. Lai, "Tensor-based projection using ridge regression and its application to action classification," *IET Image Process.*, vol. 4, no. 6, pp. 486–493, Dec. 2010.

[36] X. Geng, K. Smith-Miles, Z.-H. Zhou, and L. Wang, "Face image modeling by multilinear subspace analysis with missing values," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 3, pp. 881–892, Jun. 2011.

[37] Q. Zhao *et al.*, "Higher order partial least squares (HOPLS): A generalized multilinear regression method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1660–1673, Jul. 2013.

[38] C.-Y. Chen and K. Grauman, "Inferring unseen views of people," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 2011–2018.

[39] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 208–220, Jan. 2013.

[40] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[41] A. Chan and N. Vasconcelos, "Modeling, clustering, and segmenting video with mixtures of dynamic textures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 909–926, May 2008.

[42] A. Davies, J. Yin, and S. Velastin, "Crowd monitoring using image processing," *Electron. Commun. Eng. J.*, vol. 7, no. 1, pp. 37–47, 1995.

[43] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[44] A. Marana, L. da Fontoura Costa, R. Lotufo, and S. Velastin, "Estimating crowd density with Minkowski fractal dimension," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1999, vol. 6, pp. 3521–3524.

[45] R. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Syst., Man Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.

[46] C. Arteta, V. Lempitsky, J. A. Noble, and A. Zisserman, "Interactive object counting," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 504–518.

[47] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.

[48] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[49] Y. Zhang, K. Chen, and H.-Z. Tan, "Performance analysis of gradient neural network exploited for online time-varying matrix inversion," *IEEE Trans. Autom. Control*, vol. 54, no. 8, pp. 1940–1945, Aug. 2009.

[50] K. Chen, S. Yue, and Y. Zhang, "Matlab simulation and comparison of Zhang neural network and gradient neural network for online solution of linear time-varying matrix equation," *Lecture Notes Comput. Sci.*, vol. 5227, pp. 68–75, 2008.

[51] C. M. Bishop, *Pattern Recognition and Machine Learning*, M. Jordan, J. Kleinberg, and B. Schölkopf, Eds. New York, NY, USA: Springer-Verlag, 2007.

[52] C. Saunders, A. Gammerman, and V. Vovk, "Ridge regression learning algorithm in dual variables," in *Proc. Int. Conf. Mach. Learn.*, 1998, pp. 515–521.

Ke Chen received the B.E. degree in automation and the M.E. degree in software engineering from Sun Yat-Sen University, in 2007 and 2009, respectively, and the Ph.D. degree in computer vision from Queen Mary University of London in 2013. He is currently a Postdoctoral Research Fellow with the Department of Signal Processing, Tampere University of Technology. He is the author of more than 40 peer-reviewed conference and journal papers in computer vision, neural computing, and robotics inverse kinematics. His research interests include computer vision, pattern recognition, neural dynamic modeling, and robotic inverse kinematics.

Joni-Kristian Kämäräinen (M'15) received the M.Sc. and Ph.D. degrees from Lappeenranta University of Technology, in 1999 and 2003, respectively. He leads the Computer Vision Group and his research focuses on 2-D and 3-D scene analysis, object detection and recognition, signal processing, and machine intelligence. He is currently an Associate Professor of signal processing with the Department of Signal Processing, Tampere University of Technology, Finland.