CROWD DENSITY ANALYSIS USING SUBSPACE LEARNING ON LOCAL BINARY PATTERN

Hajer Fradi, Xuran Zhao, Jean-Luc Dugelay

EURECOM, FRANCE
fradi@eurecom.fr, zhao@eurecom.fr, dugelay@eurecom.fr

ABSTRACT

Crowd density analysis is a crucial component in visual surveillance for security monitoring. This paper proposes a novel approach for crowd density estimation. The main contribution of this paper is two-fold: First, we propose to estimate crowd density at patch level, where the size of each patch varies in such way to compensate the effects of perspective distortions; second, instead of using raw features to represent each patch sample, we propose to learn a discriminant subspace of the high-dimensional Local Binary Pattern (LBP) raw feature vector where samples of different crowd density are optimally separated. The effectiveness of the proposed algorithm is evaluated on PETS dataset, and the results show that effective dimensionality reduction (DR) techniques significantly enhance the classification accuracy. The performance of the proposed framework is also compared to other frequently used features in crowd density estimation. Our proposed algorithm outperforms the state-of-the-art methods with a significant margin.

Index Terms— Crowd density, local binary pattern, dimensionality reduction, classification

1. INTRODUCTION

There is currently significant interest in visual surveillance systems for crowd density analysis. In particular, the estimation of crowd density is receiving much attention in security community. Its automatic monitoring could be used to detect potential risk and to prevent overcrowd (e.g. in religious and sport events). Many stadium tragedies could illustrate this problem, as well as the Love Parade stampede in Germany and the Water Festival stampede in Colombia. To prevent such mortal accidents and for safety control, crowd density estimation could be used. It is extremely important information for early detection of unusual situations in large scale crowd to ensure assistance and emergency contingency plan.

In order to address the problem of crowd density estimation, many works have been proposed so far. In this context, the classification introduced by Polus [1] is commonly adopted. Based on that, the crowd density is categorized into 5 levels: free, restricted, dense, very dense, and jammed flow. One of the key aspects of crowd density analysis is related to the crowd feature extraction. Early attempts to handle this problem generally made use of texture features. In this perspective, Marana et al. assume [2] that high density crowd has fine patterns of texture, whereas, images of low density have coarse patterns of texture. Based on this assumption, many texture features have been proposed such as: Gray Level Co-occurrence Matrix (GLCM) [2, 3], Gradient Orientation Co-occurrence Matrix (GOCM) [4] and wavelet [5]. Among these features, GLCM is probably the most frequently used, from which usually 4 statistical properties are selected (contrast, homogeneity, energy, and entropy). These statistical texture features have the limitation of giving a global information for the entire image, and that could discard local information about the crowd. Also, these features could deal with occlusions that prevalently exist in crowded scenes only to some extent. As a result, the use of local texture features, especially some variants of LBP [6], has been an active topic of recent research. For instance, in [7], LBP is used in blocks, then, Dual-Histogram LBP is applied for crowd density estimation. In [8], the dynamic texture of the walking crowd is used by extracting a sparse spatio-temporal local binary pattern (SST-LBP) feature. Afterwards, the statistical property of SST-LBP is used as crowd feature. In [9], the authors propose to compute GLCM on LBP image instead of the original gray image. Finally, in [10], a crowd density estimation approach using histogram model classification is proposed, where the histogram model is based on an improved uniform local binary pattern features.

The methods mentioned above generally perform crowd density level classification directly using the high dimensional LBP-based feature vector, which might incur at least two problems: first, the high dimensional feature vector increases the computation time; second and more important, these high dimensional feature vectors generally contain components irrelevant to crowd density, and the use of the whole feature vector without any feature selection process could lead to unsatisfactory classification performances.

The contribution of this paper is then two-fold: First, the patch level analysis step involves the estimation of patch size in the real-world coordinates with incorporation of the effects of perspective distortions on patches size. Second, instead

of using the raw LBP feature for classification, we propose the combination of Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) to find a low dimensional discriminative subspace in which same-density-level samples are projected close to each other while different-density-level samples are projected further apart. This process is favourable for the later multi-class Support Vector Machine (SVM) classification step since the influence of feature components irrelevant to crowd density is minimized.
The remainder of the paper is organized as follows: we introduce our proposed approach for crowd density estimation in Section 2. The proposed approach is evaluated using PETS dataset and the experimental results are summarized in Section 3. Finally, we briefly conclude and discuss some potential future works in Section 4.

2. PROPOSED APPROACH

In this section, our proposed approach for crowd density estimation is presented. First, patch level analysis is introduced. Then, to determine the contents of each image patch under analysis, texture features are extracted using subspace learning (or dimensionality reduction) on block-based LBP. Specifically, the feature vectors are projected into discriminant space using LDA over the PCA subspace. Afterwards, the extracted features are classified into different crowd density levels by applying SVM.

2.1. Patch Level Analysis

We propose to perform crowd density estimation at frame sub-regions, which is commonly referred as *patch level*. Crowd density at patch level is more appropriate than at frame level, since it enables both the detection and the location of potential crowded areas. Actually, in many video surveillance applications and for security reasons, not only the estimation of the crowd level is required, but also the location of the crowd within the whole frame. Moreover, estimating the crowd density at image patches enables to work within Regions of Interest (RoI). In fact, more interest is usually given to the prediction of the crowd level in some specific areas compared to others, such as in the walkways.

To assign image patches to crowd density levels, the first difficulty underlied in our paper concerns the implementation of crowd levels definition introduced by Polus [1]. It consists of defining 5 crowd levels according to the range of density. This definition has been widely used for crowd density estimation, but, the estimation of the real size (of image, or image blocks) is usually neglected in previous works. In our proposed approach, we use the camera calibration parameters [11] to transform the image coordinates to the real-world coordinates, from which we can estimate the real size of any RoI within a frame. At this stage, we also take into account the effects of perspective distortions on patch size. Similar to the real size estimation, this problem is also not studied in the literature except in [12], where an approximation of the perspective map is made by linearly interpolating the two extreme lines of the scene. The effects of perspective distortions can be simply explained by the fact that objects far away from the camera appear smaller than the closest ones [13]. Therefore, to use only one definition of crowd levels under different locations within the whole frame, the effects of perspective distortions have to be compensated on the patch sizes in such way that all the extracted patches correspond to a similar size in the real-world coordinates.

2.2. Block-based Local Binary Pattern extraction and histogram sequence normalization

Recently, LBP [6] has aroused increasing interest in many applications of image processing and computer vision, in particular, it has been extensively related to the field of face recognition. Likewise, substantial progress has been achieved over the last years in crowd density analysis using LBP. The advantage of using LBP as feature extractor is that it is a powerful descriptor that characterizes the structure of the local image texture which is highly relevant to the crowd density. LBP operator is based on labeling the pixels of an image by thresholding the 3 x 3-neighborhood of each pixel with the center value and considering the result as a binary digit. Then, a binary number is obtained by concatenating all binary values in a clockwise direction, starting from the top left neighbor. Thus, for a given pixel at (x_c, y_c) position, the LBP code in decimal form is defined as:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} S(i_p - i_c)2^p \tag{1}$$

where i_c and i_p denote, respectively, the gray values of the center pixel and the P surrounding pixels. S refers to a thresholding function defined as: $S(x) = \begin{cases} 1 & \text{if } (x \geq 0) \\ 0 & \text{otherwise} \end{cases}$
In our proposed approach, each image patch is spatially divided into several non-overlapping blocks from which LBP codes are computed. Then, histogram of each block is extracted by collecting the occurrence of LBP codes. Finally, the histogram pieces computed from different blocks are concatenated into a single histogram sequence to represent a given image patch. Assume that each image patch is divided into m blocks $B_1, B_1, .., B_m$, the histogram of each image patch is formulated as follows:

$$\begin{aligned} H &= ((h_0^1, h_1^1, ..., h_{L-1}^1), ..., (h_0^m, h_1^m, ..., h_{L-1}^m))) \\ h_l^j &= \textstyle\sum_{(x,y)\in B_j} f\{LBP(x,y) = l\} \end{aligned} \tag{2}$$

where $[0, ..., L-1]$ denotes the range of gray levels in LBP map, and f is defined as: $f\{A\} = \begin{cases} 1 & \text{if } (A \text{ is true}) \\ 0 & \text{otherwise} \end{cases}$

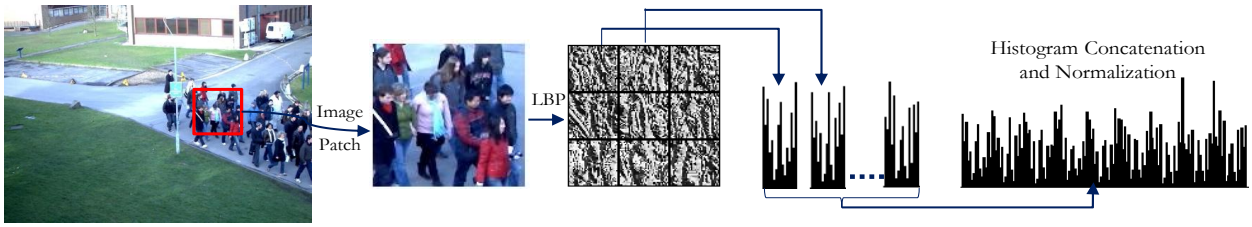

Fig. 1. *Block-based LBP extraction and Histogram sequence normalization*

Given different patch sizes, it is important to apply block normalization to each feature vector (i.e. LBP histogram sequence defined in (2)). For this purpose, $L1 - sqrt$ [14] defined as follows is used:

$$H = \sqrt{H/(|| \ H \ ||_1 + \epsilon)} \tag{3}$$

where ϵ is a small constant.
The histogram sequence defined in (3) is used as texture descriptor. An overview of the block-based LBP extraction and the histogram normalization on image patch is shown in Figure 1. For more accuracy, we resort to dimensionality reduction techniques in order to reduce the dimension of feature vector (L x m) before performing the classification step.

2.3. Discriminative subspace learning

As described in the previous section, the LBP feature vector extracted from an image patch is high-dimensional, which brought the inconvenience for the modeling and classification steps due to the so-called "curse of dimensionality". Moreover, the feature vector contains substantial amount of component dimensions which is irrelevant to the underlying crowd density and could have even a negative effect on the classification performance. One simple way to handle this problem is to apply the so called *uniform patterns* [10]. But, the use of uniformity measure has the limitation of loosing some texture information, which is not appropriate for crowd measurement. That why, we instead propose to use dimensionality reduction techniques to alleviate the effect of high-dimensional feature vector.

Linear Discriminant Analysis is a well-known, simple, but efficient approach to dimensionality reduction, and is widely used in various classification problems. It aims to find an optimized projection W_{opt} that projects D dimensional data vectors U into a d dimensional space by: $V = W_{opt}U$, in which intra-class scatter (S_W) is minimized while the inter-class scatter (S_B) is maximized. S_W and S_B are determined according to:

$$S_W = \sum_{j=1}^{c} \sum_{i=1}^{l_j} (u_i^j - \mu_j)(u_i^j - \mu_j)^T, \tag{4}$$

and

$$S_B = \sum_{j=1}^{c} l_j (\mu_j - \mu)(\mu_j - \mu)^T, \tag{5}$$

where u_i^j is the i^{th} sample of of class j, μ_j is the mean of class j, c is the number of classes, and l_j is the number of samples in class j. W_{opt} is obtained according to the objective function:

$$W_{opt} = \arg\max_W \frac{W^T S_B W}{W^T S_W W} = [w_1, \ldots, w_g] \tag{6}$$

where $\{w_i | i = 1, \ldots, g\}$ are the eigenvectors of S_B and S_W which correspond to the g largest generalized eigenvalues according to:

$$S_B w_i = \lambda_i S_W w_i, i = 1, \ldots, g \tag{7}$$

Note that there are at most $c - 1$ non-zero generalized eigenvalues, so g is upper-bounded by $c - 1$. Since $S_W$is often singular, it is common to first apply Principal Component Analysis (PCA) [15] to reduce the dimension of the original vector. This dimensionality reduction process of PCA followed by LDA is well accepted in face recognition domain and is commonly referred to as "Fisherface" [16]. In our work, we adopt the same strategy in crowd density estimation problem.

2.4. Multi-Class SVM classifier

Once the dimensionality reduction (PCA+LDA) is applied on LBP feature vectors (defined in (3)), the crowd density classification is performed by adopting SVM [17]. Since SVM is originally two-class based pattern classification algorithm, multi-class SVM classifier is constructed by combining several binary classifiers.

Let consider a training set of N pairs $(v_1, l_1), ..., (v_N, l_N)$, where $v_i \in \mathbb{R}^d$ refers to the reduced feature vector of a given image patch i, and $l_i \in \{1, ..., c\}$ is the label which indicates the crowd density level of a sample v_i. Using "One-against-one" [18], to classify an input feature vector v_i, $k(k-1)/2$ binary SVM classifications are performed and the output of all their decision functions are combined. For this purpose, "Max Wins" strategy is employed, in which the class of a given feature vector is the

one that gets the highest number of votes.
In our experiments, two types of SVM kernels are evaluated:
Linear kernel: $k(x, y) = x.y$
Radial Basis Function (RBF) kernel: $k(x, y) = e^{|x-y|^2/2\sigma^2}$

3. EXPERIMENTAL RESULTS

3.1. Dataset

The proposed algorithm is evaluated within PETS 2009 public dataset [1]. In particular, we selected some frames from S_1 and S_2 Sections. Then, we define different crowd levels [1] according to the range of people in $13m^2$, see Table 1.

Levels of Crowd Density	Range of Density ($people/m^2$)	Range of People
Free Flow	< 0.5	< 7
Restricted Flow	0.5-0.8	7-10
Dense Flow	0.81-1.26	11-16
Very Dense Flow	1.27-2.0	17-26
Jammed Flow	> 2.0	> 26

Table 1. Definition of different crowd levels according to the range of density, and according to the range of people in an area of an approximate size $13m^2$.

Actually this area ($13m^2$) corresponds to the real size of image block of size 226 x 226 (in the bottom of a frame). Then, the remaining image patches from bottom to top are carefully selected with different patch sizes according to their spatial localization in order to attenuate the effects of perspective distortions before estimating crowd levels. The extraction of multi-scale patches is shown in Figure 2.

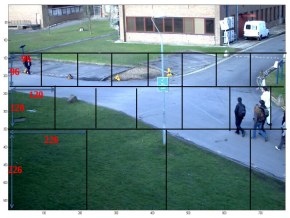

Fig. 2. *Multi-scale patches*

Afterwards, we manually labeled these image patches according to the congesting degrees of the crowd defined in Table 1. Using PETS dataset, we could not reach level 5 of the crowd (Jammed Flow), therefore, only four levels are experimented. For each crowd level, 200 image patches are selected, 100 for training and another 100 patches for testing. This results in a 4-class training set and a testing set of 400 samples each.
SVM parameters are optimized within the training set, using cross-validation (we randomly choose 20 patches to tests, for each crowd level). The same strategy was adopted for selecting PCA and KNN parameters.

[1] http://www.cvg.rdg.ac.uk/PETS2009/

3.2. Experiments

As described in Section 2.2, LBP features are extracted from 3 x 3 blocks in each patch sample, and PCA and LDA subspaces are trained with the labeled training set. The projections of training samples are further used for training multi-class SVM classifiers as described in Section 2.4. The performance is evaluated in two ways. First, for each test sample, the feature vector using block-based LBP is projected into the learned PCA and LDA subspaces, and is identified as one of the four classes by the multi-class SVM classifiers following One-against-One strategy. The top-1 identification accuracy is reported. Second, the Receiver Operating Characteristics (ROC) curve of each class is reported to demonstrate the discriminative power of our proposed feature for each crowd density level separately. Furthermore, in our experiments, both of linear and RBF SVM kernels are evaluated. Their performances are compared to K-Nearest Neighbour (KNN) classifier. We also compare our proposed feature (i.e. customized LBP) to other texture features, namely, HOG [14], Gabor wavelet [19] and GLCM [2].

3.3. Results and analysis

We first report the classification accuracy achieved by using SVM on the raw LBP features and on LBP plus dimensionality reduction techniques (LBP+PCA+LDA).

		Features Extractor	
Kernel	Classifier	LBP	LBP+PCA+LDA
Linear	One vs. One	71.00%	87.25%
RBF	One vs. One	70.00%	89.75%

Table 2. Improvement in the classification accuracy made by the dimensionality reduction on LBP feature using both linear and RBF kernels of SVM

As shown in Table 2, the classification accuracy is improved by around 20% using RBF kernel (and around 16% using linear kernel), after applying dimensionality reduction techniques over using directly raw LBP features. These results demonstrate the relevance of the discriminant feature selection process. It is also important to note that using Uniform LBP instead of LBP does not provide good results.

Obviously, a key step in crowd density estimation is the choice of texture feature. That is why, we compare our proposed feature LBP+DR (which stands for LBP+PCA+LDA) with other frequently used texture features: HOG, Gabor, and GLCM, see Figure 3.
In this Figure, we also include comparison between SVM

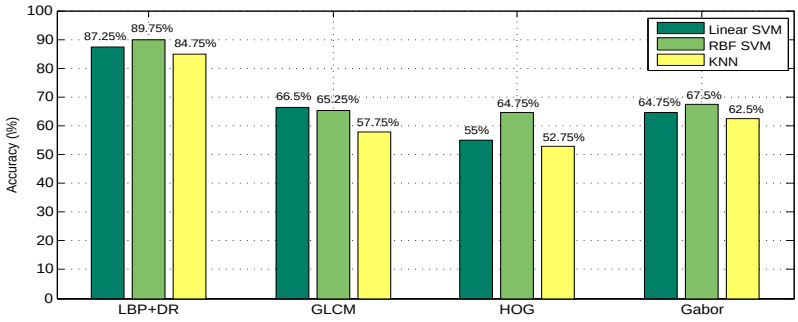

Fig. 3. *Comparisons of our proposed feature with other texture features using Linear SVM, RBF SVM, and KNN classifiers*

(for both linear and RBF kernels) and KNN classifiers. These comparisons clearly show that our proposed feature (LBP+DR) outperforms all the other texture features. In addition, the classification accuracy using SVM shows substantial improvement over KNN classification (with better results of RBF kernel compared to linear kernel). In overall, the combination LBP+DR+SVM (using RBF kernel) gives the best results in terms of classification accuracy (89.75%) with a significant margin compared to the other tested texture features. As illustrated in Figure 3, SVM classifier using RBF kernel has almost the best overall performance for all aforementioned texture features, and is thus selected for next experiments.

At this stage, we intend to evaluate the accuracy of texture features for each crowd level independently from the others; it means to explore how much each texture feature is discriminative to a specific level. To achieve this goal, ROC curve for each crowd level class is reported, see Figure 4. Then, the performance of each texture feature in a specific crowd level is measured by computing the area under the curve (AUC) and the accuracy (ACC), the results are reported in Table 3. As it shown in Figure 4 and also demonstrated in Table 3, LBP+DR outperforms all other texture features at any crowd level. Also, the results show that the tested texture features presented better discriminative ability for free and very dense flows (level 1 and level 4) compared to restricted and dense flows (level 2 and level 3). So, most of the confusions in the classification step are made in the intermediate classes, however, the results show that LBP+DR succeed to overcome this difficulty, in terms of AUC and ACC.

4. CONCLUSION

Crowd density estimation has emerged as a major component for crowd monitoring and management in visual surveillance domain. In this paper, we focus on texture analysis to characterize the crowd. In particular, we apply PCA and LDA to enhance the discriminative and descriptive power of LBP features. Furthermore, we include a large comparative study to prove that among numerous texture features only few of them are discriminative to the crowd. The experimental results highlight the role of low-dimensional compact representation of LBP on the classification accuracy. In addition, our proposed approach is robust enough to perform well in different levels of the crowd. Also, by means of comparisons with other texture features, our proposed approach has been experimentally validated showing accurate results. For future works, there is still untapped potential to reduce the complexity of the multi-classification problem. Also for tests, although jammed crowd level could not be investigated using PETS, the use of this dataset is relevant since it is well known in video surveillance community, and is publically available, thus additional comparisons could be performed. Nevertheless, we plan to use more challenging datasets as perspective.

5. REFERENCES

[1] A. Polus, J. L. Schofer, and A. Ushpiz, "Pedestrian flow and level of service," *Journal of Transportation. Engineering*, vol. 109, pp. 46–56, 1983.

[2] A. N. Marana, S. A. VelaStin, L. F. Costa, and R. A. Lotufo, "Estimation of crowd density using image processing," *IEEE Colloquium Image Processing for Security Applications*, vol. 11, pp. 1–8, 1997.

[3] K. Keung, L. Y. Xu, and X. Wu, "Crowd density estimation using texture analysis and learning," *IEEE International Conference on Robotics and Biometrics*, pp. 214–219, 2006.

[4] W. Ma, L. Huang, and Ch. Liu, "Estimation of crowd density using image processing," *Computer Sciences and Convergence Information Technology*, pp. 170–175, 2010.

[5] A. N. Marana and V. V. Verona, "Wavelet packet analysis for crowd density estimation," *IASTED International Symposia on Applied Informatics*, pp. 535–540, 2001.

[6] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classifica-

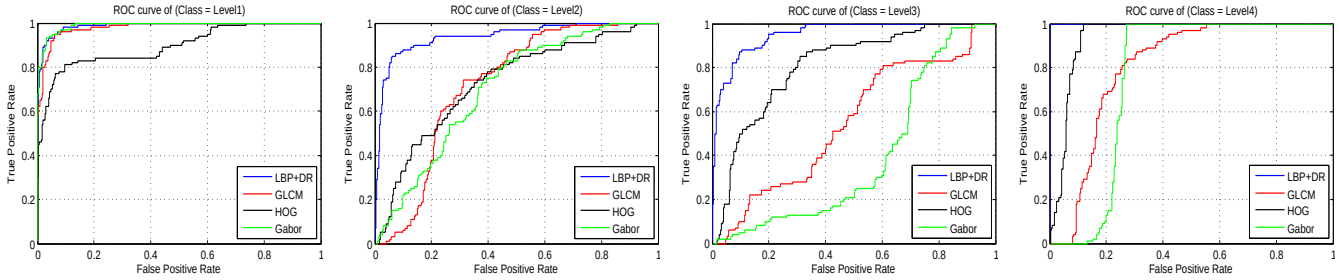

Fig. 4. *Comparisons of the ROC curves of the proposed feature (LBP+DR) with other texture features (GLCM, HOG, Gabor) for 4 different crowd levels (free, restricted, dense, very dense flows) using RBF kernel for SVM classification*

Features	Level 1		Level 2		Level 3		Level 4	
	AUC	ACC(%)	AUC	ACC(%)	AUC	ACC(%)	AUC	ACC(%)
LBP+DR	0.98	95.25	0.93	91.50	0.95	87.00	1.00	97.50
GLCM	0.98	91.75	0.72	65.75	0.55	75.00	0.80	39.50
HOG	0.89	87.75	0.72	76.00	0.80	76.25	0.94	90.25
Gabor	0.99	91.75	0.70	70.75	0.39	65.75	0.76	79.25

Table 3. Evaluation of texture features for each crowd level in terms of AUC and ACC

tion with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[7] W. Ma, L. Huang, and C. Liu, "Advanced local binary pattern descriptors for crowd estimation," *Computational Intelligence and Industrial Application*, vol. 2, pp. 958–962, 2008.

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

[9] Z. Wang, H. Liu, Y. Qian, and T. Xu, "Crowd density estimation based on local binary pattern co-occurrence matrix," *IEEE International Conference on Multimedia and Expo Workshops*, 2012.

[10] S. M. Mousavi, S. O. Shahdi, and S. A. R. Abu-Bakar, "Crowd estimation using histogram model classification based on improved uniform local binary pattern," *International Journal of Computer and Electrical Engineering*, vol. 4, pp. 256–259, 2012.

[11] Tsai and Y. Roger, "An efficient and accurate camera calibration technique for 3d machine vision," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 364–374, 1986.

[12] W. Ma, L. Huang, and C. Liu, "Crowd density analysis using co-occurrence texture features," *International Conference on Computer Sciences and Convergence Information Technology*, pp. 170–175, 2010.

[13] H. Fradi and J. L. Dugelay, "Low level crowd analysis using frame-wise normalized feature for people counting," in *IEEE International Workshop on Information Forensics and Security*, December 2012.

[14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 886–893, 2005.

[15] I. T. Jolliffe, "Principal component analysis," *2nd ed. New-York: Springer-Verlag*, 2002.

[16] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 711–720, 1997.

[17] C. Corinna and V. Vladimir, "Support-vector networks," *Machine Learning*, vol. 20, 1995.

[18] U. H.-G. Kressel, "Pairwise classification and support vector machines," *Advances in kernel methods, MIT Press, Cambridge, MA*, pp. 255–268, 1999.

[19] S. Shan, W. Gao, Y. Chang, B. Cao, and P. Yang, "Review the strength of gabor features for face recognition from the angle of its robustness to mis-alignment," *International Conference on Pattern Recognition*, 2004.