A NEW MULTICLASS SVM ALGORITHM AND ITS APPLICATION TO CROWD DENSITY ANALYSIS USING LBP FEATURES

Hajer Fradi, Jean-Luc Dugelay

EURECOM
Sophia Antipolis, France

ABSTRACT

Crowd density analysis is a crucial component in visual surveillance for security monitoring. In this paper, 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. The main contribution of this paper is two-fold: First, we propose to learn a discriminant subspace of the high-dimensional Local Binary Pattern (LBP) instead of using raw LBP feature vector. Second, an alternative algorithm for multiclass SVM based on relevance scores is proposed. The effectiveness of the proposed approach is evaluated on PETS dataset, and the results demonstrate the effect of low-dimensional compact representation of LBP on the classification accuracy. Also, the performance of the proposed multiclass SVM algorithm is compared to other frequently used algorithms for multi-classification problem and the proposed algorithm gives good results while reducing the complexity of the classification.

***Index Terms*—** Crowd density, multiclass SVM, local binary pattern, dimensionality reduction

1. INTRODUCTION

There is currently significant interest in visual surveillance systems for crowd analysis. In particular, the automatic monitoring of crowd density is receiving much attention in security community. It is extremely important information for early detection of unusual situations in large scale crowd to ensure assistance and emergency contingency plan.

One of the key aspects of crowd density analysis is crowd feature extraction. Early attempts to handle this problem generally made use of texture features. Based on the assumption [1] that high density crowd has fine patterns of texture, whereas, images of low density have coarse patterns of texture, many texture features have been proposed to address the problem of crowd density estimation such as: GLCM [1, 2], GOCM [3] and wavelet [4]. Recently, the use of local texture features has been an active topic, especially some variants of LBP [5] (e.g. Dual-Histogram LBP in [6], spatio-temporal LBP in [7], GLCM on LBP image in [8], and an improved uniform LBP in [9]).

These methods generally perform crowd density level classification directly using the high dimensional LBP-based feature vector, which might contain components irrelevant to crowd density. And the use of the whole feature vector without a feature selection process could lead to unsatisfactory classification performances. Therefore, in this paper, we propose the combination of Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) to find a low dimensional discriminative subspace where samples of different crowd density levels are optimally separated. This process is favourable for the later Support Vector Machine (SVM) classification step. In this last step, we propose a new multi-classification algorithm that involves less binary SVM classifiers by using relevance scores.

The remainder of this paper is organized as follows: we present our feature extraction block in Section 2. Then, our proposed algorithm for multiclass SVM is described in Section 3. After that, the proposed approach is evaluated using PETS dataset and the experimental results are summarized in Section 4. Finally, we briefly conclude in Section 5.

2. SUBSPACE LEARNING ON LOCAL BINARY PATTERN

In this section, our proposed feature for crowd density estimation is presented [10]. In this context, estimating crowd density is more appropriate at patch level than at frame level, since it enables both the detection and the location of potential crowded areas within the whole frame. Also, it is important to compensate the effects of perspective distortions on patch sizes in a such way that all extracted patches correspond to the same size in real-world coordinates. Then, to determine the contents of each image patch under analysis, texture features are extracted using subspace learning on block-based LBP.

2.1. Block-based LBP extraction and histogram sequence normalization

LBP [5] was originally proposed for texture analysis, and it has aroused increasing interest in many applications of image processing and computer vision. In particular, substantial progress in crowd density analysis has been achieved over the last years using LBP. 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 (x_c, y_c), 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 histogram sequences are extracted by computing the occurrence of LBP codes. Then, the histogram pieces computed from different blocks are concatenated into a single histogram sequence to represent a given image patch. So, for m blocks $\{B_1, B_2, .., B_m\}$, the histogram of each image patch is defined 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: $f\{A\} = \begin{cases} 1 & \text{if } (A \text{ is true}) \\ 0 & \text{otherwise} \end{cases}$

Given different patch sizes, block normalization to each feature vector (i.e LBP histogram sequence defined in (2)) have to be applied. For this purpose, $L1 - sqrt$ [11] defined as follow is used:

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

where ϵ is a small constant.

2.2. 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 are irrelevant to the underlying crowd density and could have a negative effect on the classification performance. So, we propose to use dimensionality reduction techniques to alleviate this effect.

LDA is an efficient approach to dimensionality reduction, it aims at finding an optimized projection W_{opt} which 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 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 PCA [12] to reduce the dimension of the original vector. This dimensionality reduction process of PCA followed by LDA is commonly referred to as "Fisherface" [13]. In this paper, we adopt the same strategy in crowd density estimation problem.

3. MULTICLASS SVM BASED ON GRADED RELEVANCE DEGREES

Once the dimensionality reduction techniques (which stands to PCA+LDA) are applied on block-based LBP, the resulting feature vectors are classified into different crowd density levels by applying SVM classifiers. In this context, the classification introduced by Polus [14] is commonly adopted. Based on that, the crowd density is categorized into 5 levels: free, restricted, dense, very dense, and jammed flow. Since crowd density estimation involves multiclass classification and SVM is originally two-class based pattern classification algorithm, several binary SVMs have to be performed to generate multiclass SVM. To handle this problem, the most frequently used techniques are: one-vs-rest, and one-vs-one, where, for a k-class problem, k, and $k(k-1)/2$ binary SVM classifiers are performed, respectively.

At this stage, we intend to improve the classification accuracy while maintaining less computational cost over the existing multiclass SVM approaches. Our proposed algorithm consists of combining $(k - 1)$ binary classifiers into a multiclass classifier. It is presented as follows:

Algorithm 1 MultiClass SVM

Input: Training set $(v_1, l_1), ..., (v_N, l_N)$
Output: Multiclass Classifier
Training:Binary SVMs and graded relevance scores
for $j = 1$ **to** $(k-1)$ **do**

- For all samples from C_1 to C_j classes, set labels to (+1) and all samples from C_{j+1} to C_k, set labels to (-1)
- Train j^{th} binary SVM
- Classify the training samples
- if $(j > 1)$, compute fuzzy scores σ_p for all training samples v_p classified as (+1) and define $(j-1)$ thresholds by splitting the curve of sorted relevance scores into equally spaced intervals.
- if $(j < k)$, compute fuzzy scores σ_n for all training samples v_n classified as (-1) and define $(k-j-1)$ thresholds by splitting the curve of sorted relevance scores into equally spaced intervals.

end for
Testing:Classification of a new sample z_l
for $j = 1$ **to** $(k-1)$ **do**

- Classify z_l by j^{th} model
- **if** (z_l is classified as (+1))
  if $(j = 1)$ $class_j(z_l) \leftarrow C_1$ else use the defined thresholds to decide $class_j(z_l)$
  else
  if $(j = k-1)$ $class_j(z_l) \leftarrow C_k$ else use the defined thresholds to decide $class_j(z_l)$
  end if

end for
The class getting the highest votes determines the instance class, if the same number of votes, the decision is made based on the relevance scores.

So, our algorithm proceeds as follows: let consider a training set of N pairs (v_1, l_1),...,(v_N, l_N), where $v_i \in R^d$ refers to the reduced feature vector of a given image patch i, and the label $l_i \in \{C_1, ..., C_k\}$ indicates its crowd density level. The basic idea is to reassess each binary SVM classifier using relevance scores. In other words, we go beyond a binary crowd subdivision by assigning different crowd levels to the classified samples. This automatic graded crowd judgments is performed using fuzzy membership score which was proposed in [15] as a measure to quickly build graded ground truths in binary labeled databases without involving manual effort.

Since a binary SVM classification aims at finding a hyperplane that optimally separates two classes in the feature space, the distance from the hyperplane can be used to measure how much a sample is representative in one class. Therefore, the decision value $f(x_s)$ of each training sample x_s is calculated, then a fuzzy score is defined as the positive/negative class posterior probability: $\sigma_s = p(y_s = sign(f(x_s))|f(x_s))$ with a parametric model based on fitting a sigmoid function:

$$\sigma_s = \frac{1}{1 + \exp(af(x_s) + b)} \tag{8}$$

where a and b parameters are adapted on the training step. According to the fuzzy relevance scores, the positive and negatives training samples of each classifier are sorted. And different thresholds are defined so that, we can re-categorize the samples in each set into different graded crowd levels.

Our proposed SVM multi-classification algorithm can be applied for any multiclass problem, where classes are related by monotonically increasing relevance degrees. Furthermore, this algorithm incurs at least two advantages: First, the computation time is decreased because only $(k-1)$ binary SVMs are performed. Second, each binary classification can be converted to multiclass classification using relevance scores σ_s.

4. EXPERIMENTAL RESULTS

4.1. Dataset

The proposed approach for crowd density estimation 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 [14] according to the range of people in an area of 13 m^2, see Table 1.

Levels of Crowd Density	Range of Density $(people/m^2)$	Range of People (for 13 m^2)
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 13 m^2.

We use the camera calibration parameters [16] to transform the image coordinates to the real-world coordinates, from which we can approximate the real size of any RoI within a frame. In particular, 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 to compensate the perspective distortions. 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 select 20 patches to tests,

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

for each crowd level). The same strategy was adopted for selecting PCA parameter.

4.2. Experiments and Results analysis

As described in Section 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. 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-vs-one technique. One-vs-one is chosen for evaluating the performance of texture features, because it has been demonstrated in literature that it gives better results compared to other multiclass SVM methods [17]. The top-1 identification accuracy using LBP plus dimensionality reduction techniques is reported in Figure 1. The performance of our proposed feature is compared to the classification accuracy achieved using SVM on the raw LBP features, and also compared to other texture features namely, HOG [11], Gabor wavelet [18] and GLCM [1].

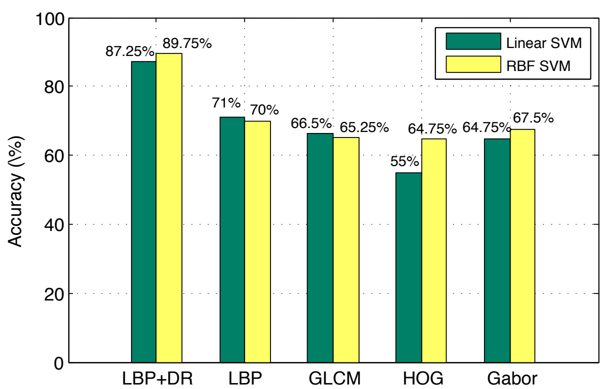

Fig. 1. *Comparisons of our proposed feature (LBP+DR) to other texture features LBP, GLCM, HOG, and Gabor for both Linear and RBF kernels using one-vs-one SVM classifier*

The comparison of our proposed feature to LBP feature demonstrates the substantial improvement made by the dimensionality reduction on LBP features in the classification accuracy. As shown in Figure 1, the classification accuracy improved 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 discriminant feature selection process. Furthermore, the comparison of our proposed feature to other frequently used texture features (HOG, Gabor, and GLCM) shows that our proposed feature (LBP+DR) outperforms all the other features. In overall, LBP+DR gives the best results in terms of classification accuracy (89.75% using RBF kernel, and 87.25% for linear SVM) with a significant margin compared to the other texture features.

At this stage, we intend to evaluate the performance of our proposed multiclass SVM algorithm based on relevance scores. To achieve this goal, the performance of LBP+DR feature using our algorithm is compared to one-vs-one and one-vs-rest methods using linear and RBF kernels.

Multiclass method	Linear SVM	RBF SVM	Number of binary SVM
One-vs-one	87.25%	89.75%	6
One-vs-rest	72.25%	84.00%	4
Proposed algorithm	88.25%	89.00%	3

Table 2. Comparisons of our proposed multiclass SVM algorithm to one-vs-one and one-vs-rest algorithms for both linear and RBF kernels using LBP+DR features

In Table 2, the classification accuracy using our proposed multiclass SVM is reported and compared to one-vs-one and one-vs-rest. We also include a comparison between these methods in terms of number of binary SVMs. According to these results, the proposed algorithm has less computational cost compared to the other multiclass SVM techniques. And its evaluation in terms of accuracy shows substantial improvement over one-vs-rest while maintaining comparable accuracy compared to one-vs-one.

5. CONCLUSION

This paper proposes a novel approach for crowd density estimation. It consists of finding 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. Specifically, LBP feature vectors are projected into discriminant space using LDA over the PCA subspace. In addition to the feature extraction block, an untapped potential to reduce the complexity of multiclass SVM problem has been explored in this paper. The alternative algorithm is based on automatic crowd judgments using relevance score, which is less computationally demanding than one-vs-one and one-vs-rest standard methods. The results show that effective dimensionality reduction techniques on LBP feature vectors significantly enhance the classification performance compared to high dimensional raw features. Also, by means of comparisons with other texture features, our proposed feature (LBP+DR) has been experimentally validated showing more accurate results with a significant margin. Furthermore, the comparison of our proposed multiclass SVM with two other standard methods highlights the usefulness of our proposed algorithm in terms of accuracy while maintaining less computational cost.

6. REFERENCES

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

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

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

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

[5] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, vol. 24, pp. 971–987.

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

[7] 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 *IEEE International Conference on Multimedia and Expo*, 2011.

[8] Z. Wang, H. Liu, Y. Qian, and T. Xu, "Crowd density estimation based on local binary pattern co-occurrence matrix," in *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops*, 2012, pp. 372–377.

[9] 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," in *International Journal of Computer and Electrical Engineering*, 2012, vol. 4, pp. 256–259.

[10] H. Fradi, X. Zhao, and J. L. Dugelay, "Crowd density analysis using subspace learning on local binary pattern," in *ICME 2013, IEEE International Workshop on Advances in Automated Multimedia Surveillance for Public Safety*, July 2013.

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

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

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

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

[15] M. Redi and B. Mérialdo, "A multimedia retrieval framework based on automatic graded relevance judgments," in *18th International Conference on Multimedia Modeling (MMM)*, 2012.

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

[17] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machines," in *IEEE Trans. Neural Networks*, 2002, vol. 13, pp. 415–425.

[18] 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," in *International Conference on Pattern Recognition*, 2004, pp. 338–341.