People Detection by Boosting Features in Nonlinear Subspace

Jie Yang, Jinqiao Wang, and Hanqing Lu

National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, China 100190
{jyang,jqwang,luhq}@nlpr.ia.ac.cn

Abstract. In this paper, we propose a novel approach to detect people by boosting features in the nonlinear subspace. Firstly, three types of the HOG (Histograms of Oriented Gradients) descriptor are extracted and grouped into one descriptor to represent the samples. Then, the nonlinear subspace with higher dimension is constructed for positive and negative samples respectively by using Kernel PCA. The final features of the samples are derived by projecting the grouped HOG descriptors onto the nonlinear subspace. Finally, AdaBoost is used to select the discriminative features in the nonlinear subspace and train the detector. Experimental results demonstrate the effectiveness of the proposed method.

1 Introduction

People detection is an important task for many video applications such as multimedia analysis and retrieval or Human-Computer Interaction. Extensive algorithms have been proposed to address the problem of people detection. The main methods being used for people detection are motion based method and shape based method. The motion based method usually take advantage of the background subtraction techniques extracting the motion foreground objects which can be further classified into different categories, e.g. human, vehicle and animal based on their shape, color, texture. These methods are influenced by the results of background modeling. The shape based method detects people directly in the image according to the shape information. The shape features include global features and local features depending on how the features are measured. One of the well-known global feature extraction method is principal component analysis (PCA). The shortcoming of the global features is that the method fails to extract discriminative features if there is a large variation in object appearance, pose and illumination. On the other hand, local features are much less sensitive to these problems. Local features descriptors have developed quickly during these years and prove to be effective on object detection and classification.

The common used local features are Haar-like wavelet features [1,2], Histograms of Oriented Gradients(HOG) features [3], region covariance [5],edgelet [6],etc. Viola [2] proposed a method which use AdaBoost and Haar-like wavelet features for pedestrian detection. AdaBoost selects a subset of these Haar-like wavelet features to form the final classifier. Dalal and Triggs [3] introduced the

G. Qiu et al. (Eds.): PCM 2010, Part I, LNCS 6297, pp. 227–235, 2010.

Histograms of Oriented Gradients (HOG) features to capture the appearance and shape feature. Zhu and Avidan [4] implemented the classifier cascade to speed up classification with HOG features obtained from variably-sized blocks.

In this paper, we propose a method to detect people by boosting features in the nonlinear subspace with higher dimension. In our method, we only detect human's head-shoulder. Because head-shoulder has the salient Ω shape which is less affected by the people's pose and the occlusions in the crowded scenes. We extract three types of the HOG descriptors from each sample and group them into one descriptor to represent the sample. The nonlinear subspace with higher dimension is constructed by using Kernel PCA and the positive and negative training samples. The final features are derived by projecting the grouped HOG descriptors onto the nonlinear subspace. AdaBoost is then employed to select the most discriminative subset of the features in the nonlinear subspace. Each selected feature corresponds to a weak classifier. The final detector is composed of all the weighted weak classifiers.

Kernel PCA was proposed as a nonlinear extension of PCA, which computes the principal components in a high dimensional feature space which is nonlinearly related to the input space. The Kernel PCA maps input space to a higher dimensional feature space, through a non-linear map, where the data is much easier separable. Therefore the high dimensional feature space contains more discriminative information than the input space. Our method enables us to exploit strengths of both Kernel PCA and AdaBoost. First, we construct the nonlinear subspace with higher dimension by using Kernel PCA, and project the grouped HOG descriptor on it to generate the final features. The generated features contain more discriminative information. Meanwhile, these features are relatively large and redundant. Then we select the most discriminative subset of these features by using AdaBoost.

2 HOG Descriptor

The histogram of oriented gradients (HOG) introduced by Dalal in [3] describes the distribution of image gradients on different orientations by a set of local histograms. HOG captures appearance and shape feature of the object and achieves good performance in pedestrian detection. The computation of a HOG descriptor is done according to the following steps:

(1) Computing gradients of the image.

The horizontal image gradient G_x and the vertical image gradient G_y are computed by filtering the image with the horizontal filter $(-1\ 0\ 1)$ and the vertical filter $(-1\ 0\ 1)^T$ respectively. The magnitude and orientation of the gradient are computed as follows:

$$N(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \tag{1}$$

$$O(x,y) = \arctan(G_x(x,y)/G_y(x,y)) \tag{2}$$

(2) Building histogram of orientation for each cell.

First, the image is divided into different cells, a cell is a spatial region like a square. Then for each cell, the histogram of the gradients is calculated by accumulating votes into bins for each orientation. The votes can be weighted by the magnitude of a gradient.

(3) Normalizing the histograms within the block.

A group of adjacent cells composes a block in which the histograms are normalized to reduce the illumination variability. In our method we choose L2-normalization.

$$v^* = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon}} \tag{3}$$

v and v^* represent the original and the normalized vector respectively, $\|v\|_2$ represents the L2-norm, ϵ is a small regularization constant preventing the denominator from being zero. The final HOG descriptor is obtained by grouping all normalized histograms into a single vector.

3 Kernel PCA

In PCA, the principle axis are obtained by diagonalizing the covariance matrix:

$$\Sigma = \frac{1}{m}\sum_{i=1}^{m} x_i x_i^T \tag{4}$$

$x^i \in R^N, i = 1, ...m$, which are centered, $\sum_{i=1}^{m} x_i = 0$ Eigenvalue equation $\Lambda = \Phi^T \Sigma \Phi$ is solved, where Φ is eigenvector matrix of Σ, and Λ is the corresponding diagonal matrix of eigenvalues. PCA can be seen as a linear projection $R^N \rightarrow R^M$ onto the lower-dimensional subspace corresponding to the maximal eigenvalues. $y = \Phi_M^T x$, where Φ_M is a submatrix of Φ containing the principal eigenvectors. y is the corresponding point of x in the linear subspace R^M.

In Kernel PCA, the nonlinear mapping is applied to the input data $\phi : R^N \rightarrow R^L$, and then solve for a linear PCA in the resulting feature space R^L, where the L is larger than M and possible infinite. We do not have to explicitly compute the nonlinear map ϕ, which can be made implicitly by using the kernel functions satisfying Mercer's theorem.

$$K(x_i, x_j) = (\phi(x_i) \cdot \phi(x_j)) \tag{5}$$

Kernel functions computes the dot product of vectors x_i and x_j in the higher dimensional space and can be thought of as functions measuring similarity between samples. The kernel value will be greater if two samples are similar. The often used kernels are Gaussian kernel, polynomial kernel and sigmoid kernel. In our method Gaussian kernel is used to compute the similarity of the samples.

$$k(x_i, x_j) = exp(-\|x_i - x_j\|^2)/\sigma^2 \tag{6}$$

Assuming that the projection of the data in feature space is zero-mean, the covariance matrix in R^L is,

$$\Sigma_K = \frac{1}{m}\sum_{i=1}^{m}\phi(x_i)\phi(x_i)^T \tag{7}$$

The eigenvalue problem becomes: $\lambda V = \Sigma_K V$. Since the eigenvector solution V must lie in the span of the training data $\phi(x_i)$, it must be true for each sample:

$$\lambda(\phi(x_i)\cdot V) = (\phi(x_i)\cdot \Sigma_K V) \qquad i = 1, ...m \tag{8}$$

And there must exist coefficients ω_i satisfying

$$V = \sum_{i=1}^{m}\omega_i\phi(x_i) \tag{9}$$

Substituting (7),(9) into (8), we can derive the equivalent eigenvalue problem formulated in terms of kernels in the input space.

$$m\lambda_i a_i = K a_i \qquad i = 1, ...m \tag{10}$$

$$m\Lambda A = KA \tag{11}$$

where $a_i = (\omega_{i,1}, ..., \omega_{i,m})^T$, $A = (a_1, ...a_m)$, and $\Lambda = diag(\lambda_1, ..., \lambda_m)$. K is a $m \times m$ matrix called Gram matrix, And each entry of this matrix $K(x_i, x_j) = (\phi(x_i)\cdot\phi(x_j))$. A is a $m \times m$ eigenvector matrix and Λ is a diagonal eigenvalue matrix with diagonal elements in decreasing order.

Since the eigenvalue equation is solved for A, we have to normalize A to make sure that eigenvalues of Kernel PCA have unit form in the feature space, so $a_i = a_i/\sqrt{\lambda_i}$. After normalization the eigenvector matrix $V = DA$ where $D = [\phi(x_1)\phi(x_2)...\phi(x_m)]$ is data matrix.

If x is a test sample whose map in the higher space is $\phi(x)$. The Kernel PCA features for x are derived as follows:

$$F = V^T\phi(x) = A^T B \tag{12}$$

where $B = [\phi(x_1)\phi(x)\phi(x_2)\phi(x)...\phi(x_m)\phi(x)]$.

4 The Proposed Method

4.1 Extracting HOG Descriptors

In our method, we implement 3 types of HOG descriptors on each sample, which capture the features at the different scales. The sample size is 32×32 pixels.

In the all these types of HOG descriptors, each block contains 2×2 adjacent cells and the adjacent blocks overlap with 2 cells. For each cell in each block, a histogram of 4 orientation bins in $0 \sim 180°$ is computed.

The other characteristics about the 3 types of HOG descriptors are describes as follows:

(1). cell size: 4×4, block size: 8×8, block number: 49. The descriptor of this type is a 784 dimensional vector.

(2). cell size: 8×8 , block size: 16×16, block number: 9. The descriptor of this type is a 144 dimensional vector.

(3). cell size: 16×16 , block size: 1×1, block number: 1. The descriptor of this type is a 16 dimensional vector.

The type (1) and type (2) HOG descriptor are used to capture the local feature of the samples, and the type (3) HOG descriptor is used to capture the global feature of the samples. We group the above three descriptors into the final HOG descriptor which is a 944 dimensional feature vector.

4.2 Using the Integral Histogram

There are much overlaps in the HOG feature computation. In the detection process, the detection windows overlap each other; In each detection window (which is a sample), the 3 types of HOG descriptors overlap each other; And in the HOG descriptor, the adjacent blocks overlap each other too. So we can use integral histogram to speed up the HOG descriptors computation.

The Integral Histogram [7] is an extension of the integral image data structure described in [1], The integral image holds at the point (x, y) in the image the sum of all the pixels contained in the rectangular region defined by the top-left corner of the image and the point (x, y). This image allows to compute the sum of the pixels on arbitrary rectangular regions by considering the 4 integral image values at the corners of the region. In order to extract histograms over arbitrary rectangular regions, we build an integral image for each bin of the histogram. By accessing these integral images, we can compute the histogram of any arbitrary rectangular regions in a short constant time.

4.3 Generating Features in Nonlinear Subspace

First, the final HOG descriptors are extracted from the training samples. Let $P_1, P_2, ..., P_u$ and $N_1, N_2, ..., N_v$ be the final HOG descriptors computed from the positive and negative training samples respectively. Then, each final HOG descriptor which is a 944 dimensional feature vector is made zero mean and unit variance. Based on the final HOG descriptor of the positive and negative training samples, We compute Gram matrices K_p and K_n for positive and negative samples, and their dimension will be $u \times u$ and $v \times v$ respectively. Eigenvector matrices A_p and A_n are calculated by solving the following eigenvalue equations:

$$u\Lambda A_p = K_p A_p$$

$$v\Lambda A_n = K_n A_n$$

where Λ and A_n are eigenvalue and eigenvector matrices, respectively.

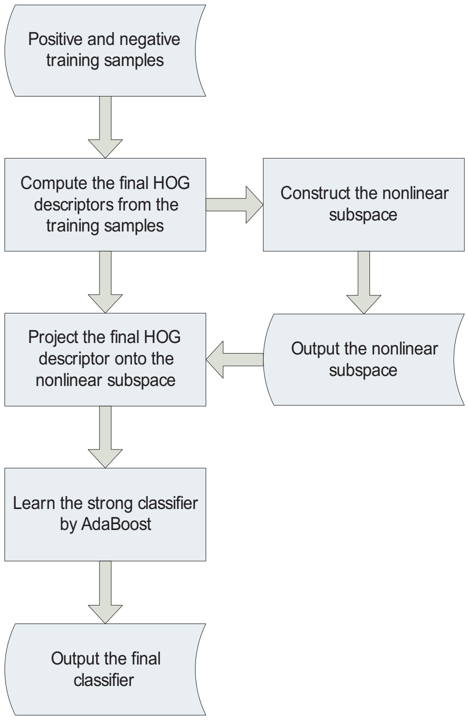


Fig. 1. The flowchart of training process in the proposed method

The Kernel PCA based feature vectors are obtained by projecting each training example onto the nonlinear subspace of the positive and negative samples respectively. By plugging A_p and A_n in eq.12, we can derive the feature vector. If f_p and f_n are the feature vectors obtained by this projection, then the final feature vector will be $f = [f_p \ f_n]$.

4.4 Learning Classifier with Boosting

The features generated by Kernel PCA lie in a nonlinear subspace with higher dimension, which contain more discriminative information than the original space. Meanwhile, these features are relatively large and redundant. We need to choose a most discriminative subset of these features. We employ AdaBoost for this purpose. AdaBoost is an ensemble learning algorithm and is effective for classification and feature selection.

We construct a weaker classifier for each dimension of the feature vector f, and use AdaBoost algorithm proposed in [11] to train the final classifier. Each dimension of the feature vector corresponds a weak classifier. The weaker classifier is a decision stump. The most N discriminative classifiers are selected by using AdaBoost. The final classifier is constructed as a weighted combination of the selected weak classifier.

The whole training process of the proposed method is described in Figure 1. After training process, we save the output classifier, the parameters of the nonlinear subspace (A_p and A_n), and the training samples. They are used to test the new samples. In order to classify a new sample, we preprocess it according to the specifications described in 4.1 and 4.3. The feature vector is obtained by projecting the final HOG descriptor onto nonlinear subspace using A_p and A_n. In eq. 12, $(x_1, x_1 ... x_m)$ are the same training samples that are used to construct the nonlinear subspace.

5 Experiments

In our experiments, we use the parts of PETS videos, INRIA pedestrian dataset and some Internet images as our training dataset. We cropped 1850 head-shoulder image patches with size of 32×32 from the training dataset. these patches and their left-right reflections are used as positive training samples. The number of total positive training samples amounts to 3700. Collecting a representative set of negative training samples is relatively arduous. We use bootstrapping method to collect the negative samples. A preliminary classifier is trained on the initial training samples, and then used to detect people on the images from the training dataset. False alarms are added to the negative training samples. We collect 3500 negative training samples in all. Typical positive training samples and negative training samples are shown in Figure 2.

Our experiments are conducted on PETS videos and some surveillance videos. We compare the performance of the detector with the features selected from the HOG descriptor feature space and the proposed method. The classifiers are trained on the same training samples. Results are shown in Figure 3 (a). The proposed method with the features boosted in the nonlinear subspace with higher dimension has better performance than the detector with the features boosted in original HOG descriptor feature space.

The Kernel PCA maps input space to a higher dimensional feature space, through a non-linear map, where the data is much easier separable. The higher dimensional feature space contains more discriminative information than the input space. The most discriminative subset of these features are selected by

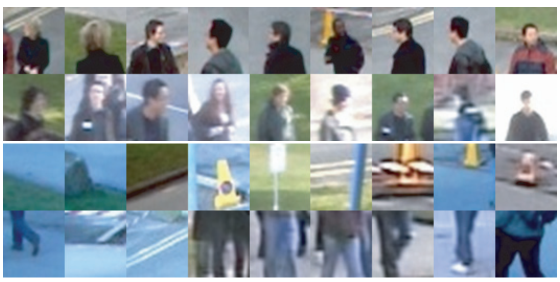
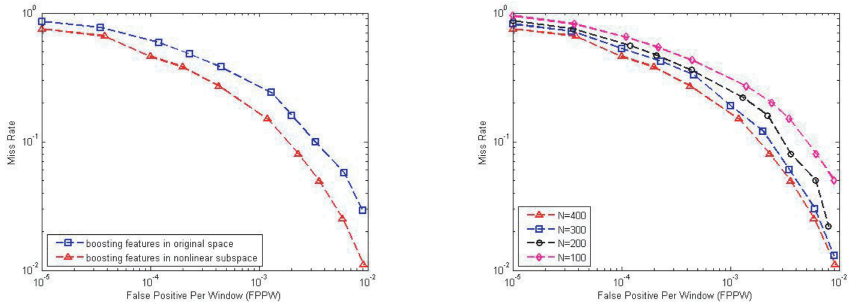


Fig. 2. Positive and negative training samples



(a) (b)

Fig. 3. Performance comparison of (a) boosting features in original space and boosting features in nonlinear subspace with higher dimension (b) the different boosted feature number N

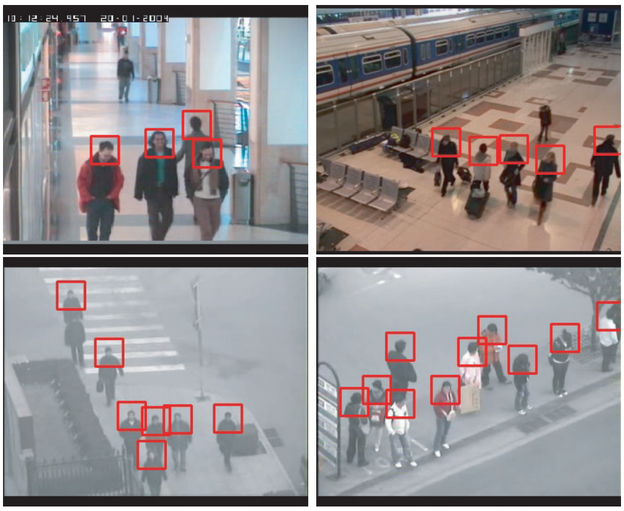


Fig. 4. Some detection results on PETS videos and surveillance videos

AdaBoost. So the features selected from the nonlinear subspace with higher dimension are more discriminative than the features in the original space.

We test the performance of the detector with the different number of the boosted feature $N = 100, 200, 300, 400$. Figure 3 (b) shows that the performance is improved when more features are selected from the nonlinear subspace. Figure 4 shows some detection results on the parts of PETS video dataset and some surveillance video dataset.

6 Conclusions

In this paper, we have proposed a novel approach for people detection by combining Kernel PCA with AdaBoost. The main contribution is mapping the grouped HOG descriptors into the nonlinear subspace where the more discriminative features are selected by AdaBoost. The Experimental results have shown the effectiveness of the proposed method.

Acknowledgments

This work is supported by National Natural Science Foundation of China (Grant No. 6060500860833006and National Basic Research Program (973) of China under contract No.2010CB327905.

References

1. Viola, P., Jones, M.J.: Rapid object detection using a boosted cascade of simple features. In: CVPR (2001)
2. Viola, P., Jones, M.J.: Detecting pedestrians using patterns of motion and appearance. In: ICCV (2003)
3. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
4. Zhu, Q., Avidan, S., Yeh, M.C., Cheng, K.T.: Fast human detection using a cascade of histograms of oriented gradients. In: CVPR (2006)
5. Tuzel, O., Porikli, F., Meer, P.: Human detection via classification on Riemannian manifolds. In: CVPR (2007)
6. Wu, B., Nevatia, R.: Detection of multiple, partially occluded humans in a single image by Bayesian combination of edgelet part detectors. In: ICCV (2005)
7. Porikli, F.: Integral histograms: a fast way to extract histograms in cartesian spaces. In: CVPR (2005)
8. Laptev, I.: Improvements of object detection using boosted histograms. In: BMVC (2006)
9. Ali, S., Shah, M.: A Supervised Learning Framework for Generic Object Detection in Images. In: ICCV (2005)
10. Moghaddam, B.: Principal Manifolds and Probabilistic Sub-Spaces for Visual Recognition. IEEE Trans. on PAMI 24(6), 780–788 (2002)
11. Freund, Y., Schapire, R.E.: Experiments with a New Boosting Algorithm. In: ICML (1996)
12. Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. The Annals of Statistics 38(2) (2000)
13. Adam, A., Shimshoni, I.: Robust Fragments-based Tracking using the Integral Histogram. In: CVPR (2006)