Optimized Cascade of Classifiers for People Detection Using Covariance Features

Malik SOUDED[1,2] and Francois BREMOND[1]
[1]*STARS team, INRIA Sophia Antipolis - Méditerranée, France*
[2]*Digital Barriers France, Sophia Antipolis, France*
{*Malik.Souded, Francois.Bremond*}*@inria.fr, Malik.Souded@digitalbarriers.com*

Keywords: People detection, Covariance descriptor, LogitBoost.

Abstract: People detection on static images and video sequences is a critical task in many computer vision applications, like image retrieval and video surveillance. It is also one of most challenging task due to the large number of possible situations, including variations in people appearance and poses. The proposed approach optimizes an existing approach based on classification on Riemannian manifolds using covariance matrices in a boosting scheme, making training and detection faster while maintaining equivalent performances. This optimisation is achieved by clustering negative samples before training, providing a smaller number of cascade levels and less weak classifiers in most levels in comparison with the original approach. Our work was evaluated and validated on INRIA Person dataset.

1 INTRODUCTION

Person detection is one of the most challenging task in computer vision. The large variety of people poses and appearences, added to all external factors like different points of view, scenes content and partial occlusions make this issue complicated. The importance of this task for many applications like people tracking especially in crowded scenes has motivated many researches. A lot of approaches were proposed as results of these research.

The most frequent scheme consists in using descriptors to modelize a person. To perform this modelization, an offline learning step is carried using these descriptors. Once the learning achieved and the classifier obtained, the detection is performed by testing all possible image subwindows.

In (Papageorgiou and Poggio, 2000), Papageorgiou and Poggio used Haar-like features to train a SVM classifier. Viola et al. (Viola et al., 2006) trained a cascade of Adaboost classifiers using Haar-like features too. In (Dalal and Triggs, 2005), a new descriptor called Histogram of Oriented Gradients (HOG) was introduced by Dalal and Triggs and was used to train a linear SVM, providing very good people detector. Dala et al. associate this descriptor later in (Dalal et al., 2006) with histograms of differential optical flow features outperforming the previous approach.

Mu et al. have proposed new variants of well known standard LBP descriptor in (Mu et al., 2008). They demonstrated the effectiveness of their Semantic-LBP and Fourier-LBP features in comparison to standard LBP for people detection. In (Schwartz et al., 2009), Schwartz et al. concatained HOG, color frequency and co-occurence matrices as one descriptor and employed Partial Least Square analysis for dimensionality reduction.

All the previous mentioned approaches can be classified as dense representation methods due to the detection method (dense search on images). Some other approaches use different detection method and can be categorized as sparse representation approaches. They consists of modeling the human body parts, detect them and achieve people detection using geometric constrains. In (Mikolajczyk et al., 2004), dedicated Adaboost detectors were trained for several body parts. The final detection is obtained by optimizing the likelihood of part occurence along with the geometric relation.

Recently, Tuzel et al. (Tuzel et al., 2007) propose a performant approach for people detection. Their approach uses covariance descriptors to characterise people. This characterisation is achieved by training a cascade of classifiers on a dataset containing human and non-human images. The training is done using a modified version of LogitBoost algorithm to deal with some specificities and constraints of covariance descriptors. One of the main covariance descriptor

issues is the computing time of all related operators and metrics. It makes covariance descriptors difficult to use for real-time processing.

Yao and Odobez (Yao and Odobez, 2008) have proposed three main contributions to this approach improving the efficiency of the training and detection stage, and providing better performances.

Because our approach is mainly based on these two last approaches, a brief recall about covariance descriptor computation, and a summary of the two cited approaches are described below.

The two approaches present some issues. In the next section, the main contribution of this paper, addressing these issues, is presented. Finally, the experimental results demonstrate that the proposed approach provides equivalent performances that the original ones while improving the processing time of the training and detection stage.

2 PEOPLE DETECTION USING COVARIANCE FEATURES

2.1 Region Covariance Descriptor

Region covariance descriptor is a powerful way to encode a large amount of information inside in a given image region. Il allows the encapsulation of a large range of different features in a single structure, representing the variances of each feature and the correlation between features.

Let I be an image of dimension $W \times H$. We can extract at each pixel location $\mathbf{x} = (x,y)^T$ a set of d features such as intensity, color, gradients, filter responses, etc.

For a given rectangular region R of I, let $\{z_i\}_{i..S}$ be the d-dimensional feature points inside R. The region R is represented with the $d \times d$ covariance matrix of the feature points

$$C_R = \frac{1}{S-1}\sum_{i=1}^{S}(z_i - \mu)(z_i - \mu)^T \qquad (1)$$

where μ is the mean of the points z_i and S the number of pixels within R

2.1.1 Used Features

In (Tuzel et al., 2007), Tuzel et al. use a 8-dimensional set of features

$$\left[x\ y\ |I_x|\ |I_y|\ \sqrt{I_x^2 + I_y^2}\ |I_{xx}|\ |I_{yy}|\ arctan\frac{|I_x|}{|I_y|}\right]^T \qquad (2)$$

where x and y are the pixel location, $I_x; I_{xx}; \ldots$ are intensity derivatives, and the last term is the gradient orientation.

Yao and Odobez (Yao and Odobez, 2008) replace the two second derivatives features $|I_{xx}|$ and $|I_{yy}|$ by two foreground measures G and $\sqrt{G_x^2 + G_y^2}$. G denotes the foreground probability value (a real number between 0 and 1 indicating the probability that the pixel x belongs to the foreground), and G_x and G_y are the corresponding first order derivatives. These foreground features are obtained using a background subtraction technique which is restricted to moving people. These two features improve people detection performances and processing time in video sequences.

2.1.2 Fast Covariance Descriptor Computation

A large number of covariance descriptors are required to achieve the training of classifier cascade and for an effective process. The computation of all the feature sums, means and variances for each region has a high cost in term of processing time. To deal with this, Integral images are ideally suited to minimize the number of numerical operations.

Integral images are intermediate image representations used for the fast calculation of region sums (Simard et al., 1998), (Viola and Jones, 2001). Each pixel of the integral image is the sum of all the pixels inside the rectangle bounded by the upper left corner of the image and the pixel of interest.

Due to the symmetic nature of covariance matrices, only upper (or lower) triangle values are needed. In the case of 8-feature set, the covariance matrix will contain 36 different values, and 44 Integral Images are computed to speed up the computing process (8 integral images for the representation of each feature independently and 36 for the representation of product for each pair of features).

2.1.3 Covariance Normalisation

In order to enhance covariance descriptors robustness toward local illumination changes, a normalization step is performed on the covariance matrix. Let r be a subregion contained in a larger region of interest R.

First, both covariance matrices C_r and C_R are computed using integral representation. After that, the values of covariance matrix C_r are normalized with respect to the standard deviations of their corresponding. features inside the detection window R as in (Tuzel et al., 2007) The normalized covariance descriptor is denoted $\hat{C}_r$.

2.1.4 LogitBoost Algorithmon Riemannian Manifolds

The classification process is performed using a cascade of classifiers which is trained using a LogitBoost algorithm on Riemannian Manifolds.

Standard LogitBoost algorithm on vector spaces: As seen in (Friedman et al., 2000), let $\{(x_i, y_i)\}_{i=1...N}$ be the set of training samples, with $y_i \in \{0,1\}$ and $x_i \in R^n$. The goal is to find a decision function F which divides the input space into the 2 classes. In LogitBoost, this function is defined as a sum of weak classifiers, and the probability of a sample x being in class 1 (positive) is represented by

$$p(x) = \frac{e^{F(x)}}{e^{F(x)} + e^{-F(x)}}, \quad F(x) = \frac{1}{2}\sum_{l=1}^{N_L} f_l(x). \qquad (3)$$

The LogitBoost algorithm iteratively learns the set of weak classifiers $\{f_l\}_{l=1...N_L}$ by minimizing the negative binomial log-likelihood of the training data:

$$-\sum_{i}^{N}[y_i log(p(x_i)) + (1-y_i) log(1-p(x_i))], \qquad (4)$$

through Newton iterations. At each iteration l, this is achieved by solving a weighted least-square regression problem: $\sum_{i=1}^{N} w_i \| f_l(x_i) - z_i \|^2$, where $z_i = \frac{y_i - p(x_i)}{p(x_i)(1-p(x_i))}$ denotes the response values, and the sample weights are given by $w_i = p(x_i)(1 - p(x_i))$.

LogitBoost algorithm on Riemannian manifolds:
To train classifiers using covariance descriptors, this algorithm is not usable as it is. In fact, covariance descriptors do not belong to vector spaces but to the Riemannian manifold $\mathcal{M}$ of $d \times d$ symmetric positive definite matrices Sym_d^+.

Based on an invariant Riemannian metric on the tangent space defined in (Pennec et al., 2006), let $\mathbf{X}$ and $\mathbf{Y}$ be two matrices from Sym_d^+, the following operators are defined and used to achieve training using LogitBoost on Riemannian manifold:

$$\exp_{\mathbf{X}}(y) = \mathbf{X}^{\frac{1}{2}} \exp(\mathbf{X}^{-\frac{1}{2}} y \mathbf{X}^{-\frac{1}{2}}) \mathbf{X}^{\frac{1}{2}} \qquad (5)$$

$$\log_{\mathbf{X}}(y) = \mathbf{X}^{\frac{1}{2}} \log(\mathbf{X}^{-\frac{1}{2}} y \mathbf{X}^{-\frac{1}{2}}) \mathbf{X}^{\frac{1}{2}} \qquad (6)$$

$$d^2(\mathbf{X}, \mathbf{Y}) = trace\left(\log^2(\mathbf{X}^{-\frac{1}{2}} \mathbf{Y} \mathbf{X}^{-\frac{1}{2}})\right) \qquad (7)$$

which are respectively the exponential, the logarithm and the squared distance on Sym_d^+ matrices.

$\exp(y) = U \exp(D) U^T$ and $\log(y) = U \log(D) U^T$. $y = UDU^T$ is the eigenvalue decomposition of the symmetric matrix y and $\exp(D)$ and $\log(D)$ are obtained by applying exponential and logarithm functions respectively on the diagonal entries of the diagonal matrix D.

Tuzel et al. have introduced a modifications to the original LogitBoost algorithm to specifically account for the Riemannian geometry. This was done by introducing a mapping function h projecting the input covariance descriptors into the Euclidian tangent space at a point μ_l of the manifold $\mathcal{M}$:

$$h(\mathbf{X}) = vec_{\mu_l}(\log_{\mu_l}(\mathbf{X})) \qquad (8)$$

where: $vec_{\mathbf{X}}(y) = vec_I(\mathbf{X}^{-\frac{1}{2}} y \mathbf{X}^{-\frac{1}{2}})$, and $vec_I(y) = [y_{1,1}\ \sqrt{2}y_{1,2}\ \sqrt{2}y_{1,3}\ ...\ y_{1,2}\ \sqrt{2}y_{2,3}\ ...\ y_{d,d}]^T$.

The trained cascade consists of a list of ordered strong classifiers. Each strong classifier contains a set of weak classifiers. A weak classifier is defined by a sub-region of interest, the corresponding mean value of covariance descriptors of all positive samples and a regression function.

To train a level k of the cascade, a given number of weak classifiers are successively added. To add a weak classifier l to the current training classifier, 200 candidate weak classifiers are evaluated: 200 subwindows are randomly selected. Let r_i be one of these subwindows and $\hat{C}_{r_i}^j$ the corresponding normalized covariance descriptor on the sample j. For each subregion r_i, the mean μ_i of all the normalized covariance descriptors $\hat{C}_{r_i}^j$ of the positive samples is computed using a gradient descent procedure described in (Pennec et al., 2006). Using this mean μ_i, all $\hat{C}_{r_i}^j$ of all the samples are projected onto the tangent space using (8) obtaining vectors in Euclidean space. Using these vectors and the corresponding weights of all samples, a regression fucntion g_i is computed.

The best weak classifier, which minimizes negative binomial log-likelihood (4), is added to the current training classifier. The weights and the probabilities of all the samples are updated according to the new added weak classifier. The positive and the negative samples are sorted in a decreasing order using their probabilities. The current strong classifier is concidered as fully trained if the difference between the probability of the $(99.8\%)_{th}$ positive sample and the $(35\%)_{th}$ negative sample is greater than 0.2.

In this case, the training of the current cascade level is achieved. The negative samples are tested with the new cascade and all correctly classified samples are removed from the training dataset. The next cascade level is trained using remaining negatives.

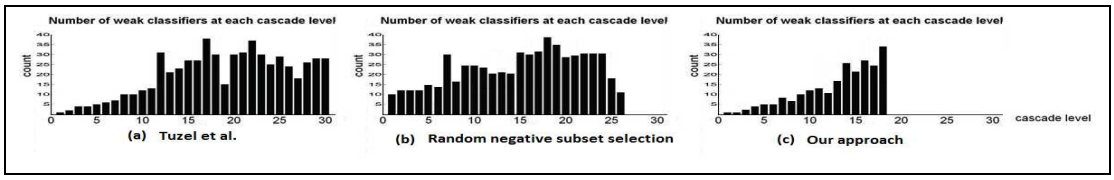

Figure 1: Comparison between structures of the cascade of classifiers: (a) Tuzel et al.(Tuzel et al., 2007); (b) trained with a random selection of negative subset; (c) our proposed approach: less cascade levels with less weak classifiers per level

Note that Yao et. al have introduced two important improvements. First, classifiers are trained on a lower dimension. They proposed an approach to select the best subset of d' features from the d original ones for each sub-region. They train classifiers on 4-feature covariance descriptors. Second, they have concatained the mean feature vector of each random sub-region to the mapped vector of each sample before regression computing, improving performances.

2.2 Main issues

The initial approach proposed by Tuzel et al. (Tuzel et al., 2007) outperforme existing approaches of the state of the art by providing a lower rate of miss-detections and false positives, but it has the disadvantage of being highly time consuming for the detection process and not applicable for real-time processing. In (Tuzel et al., 2007), Tuzel et al. indicate that detection time on a 320 x 240 image is approximatively 3 seconds for a dense scan, with 3 pixel jumps vertically and horizontally. Note that training time is relatively long also (2 days in (Tuzel et al., 2007)).

The most computationnally expensive operation during the training and the classification is eigenvalue decomposition. This decomposition is the basis of all operators in Sym_d^+. Eigenvalue decomposition of a symmetric $d \times d$ matrix requires $O(d^3)$ arithmetic operations, so computing time increases quickly by using more features.

The feature subset selection approach, proposed by Yao and Odobez in (Yao and Odobez, 2008) allows to work in a lower dimensional symmetric positive definite matrices, making eigenvalue decomposition faster and thereby improving all the training and classification processes.

We focus in our work on another way to make the classification faster while maintaining high classification performances. At the end, the obtained approach improve also the training stage.

3 Proposed Algorithm

3.1 Ordering negative training sample

Using large number of samples for the training process makes it very slow. Of course, the larger the training dataset is, the more performant the classifier cascade is. But most of the time, large number of negative samples contain very similar information. This problem is more frequent for the first cascade levels, where a new level is trained using false positives of previous levels. These false positives are generally resulting from successive small shifts of testing window on the image, providing very similar content.

One can suppose that using a smaller subset of randomly selected negative samples to train a given cascade level can be a good solution to speed up training. We can suppose that a randomly selected subset can be statistically representative of all remaining negatives.

We have tested this approach and we have observed that it effectively speeds up training and provides a lower number of cascade levels than (Tuzel et al., 2007) but with longer classifiers (See Figure. 1, b), slowing down in comparison to the previous mentioned approaches. In fact, one cascade level consists of a set of weak classifiers. The response of one classifier is obtained after computing the output values of all its weak classifiers. It means than a long classifier containing a large number of weak classifiers takes more time to return a decision, so a cascade of long classifiers is very slow for detection.

The number of weak classifiers per cascade level depends mainly on the diversity of negative samples used for the training. The characterisation of positives and their separation from negativesl requires as many subregions of interest as the samples are diverse.

To illustrate the relationship between negative samples diversity and classifiers cascade structure, let us use a simple example which can be generalized to understand the concept. Suppose that we have to separate a person image from three non-person images: a sky image, a vertical barrier image and a lamppost image. Due to the poverty of texteures and gradiends

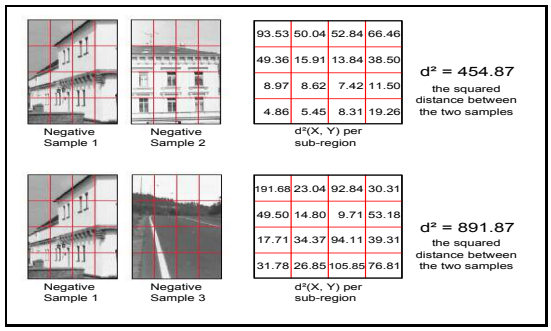

Figure 2: Used Squared distance computation between two negative samples for hierarchical clustering

on the sky image, a unique large covariance region is sufficient to separate the sky image from the person image, which has many gradients and a vertical shape. For vertical barriers, the previous region is not appropriate due to the vertical shape of the person. A smaller region around the person's head is more appropriate. The circular shape of the head provides a good separation. Now, for the lamppost, the two previous regions are not suitable. It is necessary to take a region on legs or around shoulders to encode curvatures. For this example, there are two methods to train the classifier. The first cascade is trained with one negative image at a time in the mentionned order. The second cascade is trained with the three negative images at the same time, using approriate parameters. The first method provides a cascade with three levels, each level containing one weak classifier corresopnding to one case (low texture level, vertical shapes only, circular shape at the top of vertical shape). The second method provides a cascade of a unique classifier containing three weak classifiers at least (the number can be larger du to the possible combinations).

Suppose now that we have to perform a people detection on a large image which contains only sky, a low textured road, some vertical barriers and some lampposts. Both cascades will provide equivalent detection performances, but the first one will be faster. This is because most of tested windows (sky and road) are rejected after evaluating only one covariance descriptor (the one of the first cascade level), while the second classifier cascade needs to evaluate three (or more) covariance descriptors for each tested window.

The average number of evaluated covariance descriptors using Tuzel et al. cascade (a) is 8.45 while the cascade in (b) needs more than 21 covariance descriptor evaluation.

We propose an approach using a shorter subset of negatives at each cascade level training to make it faster. Our approach provides shorter cascade with smaller classifiers on average (Figure. 1, c) in comparison with the Tuzel et al. one (Figure. 1, a) making detection process faster. At the same time, the experimental results show that our approach provides similar detection performances than the original one.

3.2 Clustering on negative data

The idea consists in regrouping negative samples per groups containing similar contents in terms of covariance information, and train each cascade level by one group of similar samples. The previously described Logitboost algorithm achieves characterization of people against a group of negative samples faster when these negative samples are more similar. It also specialize each cascade level. This negative samples regrouping is achievd using clustering methods. We tried two methods to perform this clustering, the first one is applyed directely in Riemannian manifold while the second one is performed in the Euclidean space.

3.2.1 Hierarchical clustering on covariance descriptors using covariance matrices distance

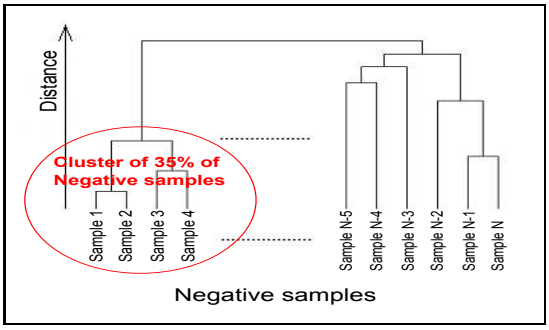

Figure 3: Hierarchical tree of clustered negative samples

A matrix containing the distances between all pairs of negative samples is computed. To compute the distance between two negative samples using covariance descriptors, each sample image is divided into a grid of 16 equal sub-regions. The final distance between the two samples is the sum of the 16 distances between each covariance descriptors pair using (7) (See Figure. 2). Once the distance matrix computed, a hierarchcal clustering is performed, providing a tree of negative samples (See Figure. 3)

3.2.2 Clustering in projection space

The second clustering approach consists in projecting all negative samples to a tangent space. In this method, we use covariance descriptor of the whole

image of each sample. The mean of all negative samples is computed and used to project all covariance descriptors to the Euclidean space. Finally Euclidean space, the clustering is performed using adaptive bandwidth mean shift filtering (Comaniciu and Meer, 2002). (See Figure. 4)

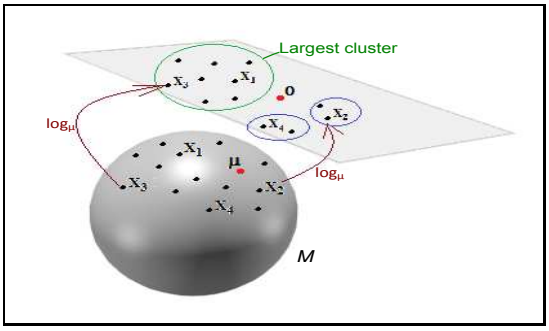

Figure 4: Clustering on tangent space

3.3 Train iteratively by each subset

After clustering, it is now possible to select the *n* most similar negatives samples. In the case of hierarchical clustering, we select first the cluster which contain *n* = 35 % of remaining negatives. we took this value according to Tuzel et al. parameters (Tuzel et al., 2007), to have similar conditions and to be able to compare results. In the mean shift clustering on tangent space, we select the cluster containing the largest number of samples. This is motivated by the desire to eliminate the largest percentage of negative samples as soon as possible.

The training is now done by applying a clustering step on the negative samples, selecting the most similar negatives subset and use it for training. After achieving current level training, the new cascade is applyed to all the remaining negative samples, those used for training and the others. We observed that 80% to 95% of the negatives from the used subset are correctly classified and removed and a small part of unused negatives too.

The clustering is repeated on remaining negatives to train next levels.

4 Experimental Results

We conduct experiments on INRIA dataset to be able to compare our results with those of Tuzel et al. (Tuzel et al., 2007).

The INRIA person dataset (Dalal and Triggs, 2005) is devided to two subsets: a training set containing 2416 person annotations and 1218 person-free images and a test dataset with 1132 persons ans 453 person-free images. This dataset is quite challenging due to the various scenes, content, and persons appearence and poses.

4.1 Detection performances comparison

The detection performances were evaluated on two criteria: miss detection rate, given by $\frac{FalseNeg}{FalseNeg+TruePos}$ and false positive per window, which is given by $\frac{FalsePos}{TrueNeg+FalsePos}$. The rightmost curve points of our method corresponds to the results of the 8 first levels of the cascade. The other points are added every 4 cascade levels. The curve in Figure 5 show that our approach provides very close performances to Tuzel et al. ones, which outperform Dalal et al. results (Dalal et al., 2006)

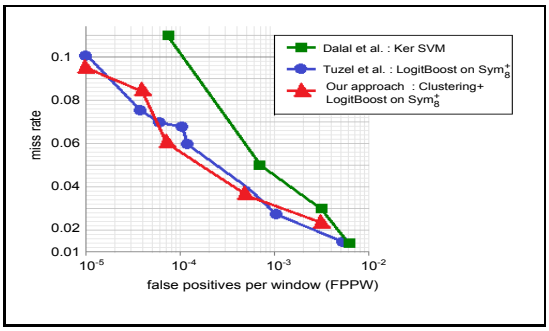

Figure 5: Comparison with the methods of Dalal et al. (Dalal et al., 2006) and Tuzel et al. (Tuzel et al., 2007) on the INRIA data set.

4.2 Classifier cascade structure, training time and detection time comparison

Our cascade of classifier (See Figure. 1 (c)) is shorter than the Tuzel et al. one. It contains 18 levels achieving rejection of more than 99% of negatives during training. Most levels contain less weak classifiers also. The average number of evaluated covariance descriptors using our cascade is 6.85 while it is 8.45 for Tuzel et al. cascade

The main consequence of this difference of structures is the detection time. Our cascade perform detection faster than the Tuzel el al. one. We have implemented both approaches with C++ and performed training and detection on an Intel(R) Core(TM) i7-920 Processor at 2.66-Ghz with 4Gbytes or RAM. The average time of detection on images of 320x240 is approximatively 2.3 seconds for Tuzel et al. while it is approximatively 0.5 seconds for our method.

In the same conditions, training time is also decreased by our approach. The training takes 22 hours for Tuzel et al. approach while it takes 9 hours for our approach.

Note finally that the clustering in tangent space provide better results for first cascade levels training, but after few levels, it becomes less precise. This can be explained by the fact that at the first level,negative samples are densly regrouped. The computed mean for tangent space projection is significant. After few level training, and removing correct classified negatives, the remaining negatives became sparse and computing a mean on sparse samples make it less signicative, the projection to tangent space is not suitable.

5 Conclusions

We have proposed an approach to optimize people detection using covariance descriptors. This approach consists in clustering negative data before training to obtain better classifier structure. The resulting detector is faster that original one and was trained in shorter time. The experimental results on a challenging dataset validate our approach.

REFERENCES

Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Machine Intell., 24:603619.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In IEEE Conf. Comp. Vision and Pattern Recognition (CVPR).

Dalal, N., Triggs, B., and Schmid, C. (2006). Human detection using oriented histograms of flow and appearance. In Europe Conf. Comp. Vision (ECCV), volume II, pages 428441.

Friedman, J., Hastie, T., and Tibshira, R. (2000). Additive logistic regression: a statistical view of boosting. Ann. Statist., 23(2):337C407.

Mikolajczyk, K., Schmid, C., and Zisserman, A. (2004). Human detection based on a probabilistic assembly of robust part detectors. In Europe Conf. Comp. Vision (ECCV), volume I, pages 6981.

Mu, Y., Yan, Y., Liu, Y., Huang, T., and Zhou, B. (2008). Discriminative local binary patterns for human detection in personal album. In CVPR 2008, pages 18.

Papageorgiou, P. and Poggio, T. (2000). A trainable system for object detection. Int. J. of Computer Vision, 38(1):1533.

Pennec, X., Fillard, P., and Ayache, N. (2006). A riemannian framework for tensor computing. Int. Journal of Comp. Vision, 66(1):4166.

Schwartz, W., Kembhavi, A., Harwood, D., and Davis, L. (2009). Human detection using partial least squares analysis. In ICCV.

Simard, P., Bottou, L., Haffner, P., and LeCun, Y. (1998). Boxlets: A fast convolution algorithm for signal processing and neural networks. Proc. Conf. Advances in Neural Information Processing Systems II, pp. 571-577.

Tuzel, O., Porikli, F., and Meer, P. (2007). Human detection via classification on riemannian manifolds. In IEEE Conf. Comp. Vision and Pattern Recognition (CVPR).

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR 01), vol. 1, pp. 511-518.

Viola, P., Jones, M., and Snow, D. (2006). Detecting pedestrians using patterns of motion and appearance. In Europe Conf. Comp. Vision (ECCV), volume II, pages 589600.

Yao, J. and Odobez, J. (2008). Fast human detection from videos using covariance feature. In: ECCV 2008 Visual Surveillance Workshop.