

A Re-evaluation of Pedestrian Detection on Riemannian Manifolds

Diego Tosato¹, Michela Farenzena¹, Marco Cistani^{1,2}, Vittorio Murino^{1,2}

¹Dipartimento di Informatica, University of Verona, Italy

²Istituto Italiano di Tecnologia (IIT), Genova, Italy

{name.surname}@univr.it

Abstract—Boosting covariance data on Riemannian manifolds has proven to be a convenient strategy in a pedestrian detection context. In this paper we show that the detection performances of the state-of-the-art approach of Tuzel et al. [7] can be greatly improved, from both a computational and a qualitative point of view, by considering practical and theoretical issues, and allowing also the estimation of occlusions in a fine way. The resulting detection system reaches the best performance on the INRIA dataset, setting novel state-of-the-art results.

Keywords—Pedestrian Detection, Riemannian Manifolds, Boosting

In Computer Vision, detecting people in images is a crucial yet hard task; this is due to the presence of many acquisition settings and the large variations of human appearance and pose. Across the many nowadays techniques, whose recent samples are [6], [7], [8], the ensemble-of-features based methods [3], [7] are very promising. They are based on boosting [4], which is reckoned to be the best way to pursue a robust classifier.

Here we focus on the state-of-the-art method in [7], where a human is modeled by covariance matrices of image features. This representation is convenient: covariances allow a great robustness, regarding for example the number of elements employed for its calculus, i.e., the size of the pedestrian. The method reaches the best performances on the INRIA dataset¹, but the price for such tools is the computational burden. In fact, covariance matrices, which belong to Sym^+ (the group of symmetric positive definite matrices), live in a Riemannian Manifold. This implies a high effort to compute all the operators needed in the boosting framework.

In this paper we propose a set of improvements, that tackle the framework of [7] under both a theoretical point of view, managing the Riemannian geometry in a finer and economic way, and a practical point of view, suggesting tricks that lead to a more robust and faster detection framework, also able to finely model occluded individuals.

I. BINARY CLASSIFICATION ON RIEMANNIAN MANIFOLDS

We first briefly introduce the LogitBoost on Riemannian manifold of [7], inheriting their notation. Let

$\{\mathbf{X}_i, y_i\}_{i=1, \dots, N}$ be the set of training examples of fixed size, with labels $y_i \in \{1 = \text{human}, 0 = \text{other}\}$ and $\mathbf{X}_i \in \mathcal{M}$, the Riemannian manifold of covariance matrices. The goal is to find a strong classifier $F(\mathbf{X}_i) : \mathcal{M} \mapsto \{0, 1\}$, formed by an ensemble of weak learners (WL), that partition the input space into 2 classes, according to the labeling. The probability of \mathbf{X}_i being in class 1 is represented by

$$p(\mathbf{X}_i) = \frac{e^{F(\mathbf{X}_i)}}{e^{F(\mathbf{X}_i)} + e^{-F(\mathbf{X}_i)}}, \quad (1)$$

where $F(\mathbf{X}_i) = \sum_{l=1}^{N_l} f_l(\mathbf{X}_i)$, and $\{f_l\}_{l=1, \dots, N_l}$ is the set of WL. Iteratively, each WL is selected by fitting a weighted least-square regression function f_l of training points \mathbf{X}_i to response values z_i and weights w_i :

$$f_l = \arg \min_{f \in \mathcal{F}} \sum_{i=1}^N w_i |z_i - f(\mathbf{X}_i)|^2. \quad (2)$$

where \mathcal{F} is the set of possible WLs. Note that each WL f_l focuses on a patch b_l , whose size and position over all data is selected by evaluating a bunch of candidate sizes and positions, sampled uniformly over the pedestrian image.

II. IMPROVEMENTS

A. Towards a faster, more informative WLs selection

We propose to speed-up the process of selection of the patches b_l , by sampling only over *interesting* position values, i.e., those pixels representing people with higher chance. This translates in exploiting a prior map of human appearance. We build a subset of positive samples, masking with 1 the pedestrian and 0 the background, and we compute the per-pixel mean. Normalizing the result by the number of positives provides the prior map. The add-on goes beyond the mere speed-up. In fact, it minimizes the selection of patches on the background area, that can be discriminative in an erroneous way. For example, if the positive dataset depicts people with a similar background, whose visual layout differs from the content of the negative dataset, the background information is very discriminative and it will be selected by the WLs. This makes a classifier not capable of generalizing with different backgrounds. This optimization is referred as OPT 1.

¹<http://pascal.inrialpes.fr/data/human/>

B. Avoiding the overtraining

Building the negative set $\{B_i\}_{i=1,\dots,N_{BG}}$ is very compelling, being it representative of everything but humans. The B_i negative samples are fed gradually and randomly into the training of the classifier, exploiting the classic boosting cascade structure that allows a very fast classification. Considering these facts we devise a strategy for the use of the negative samples. We order them by *difficulty*: *easy* negatives are clearly different to humans, while *hard* negatives are not. Experimentally, we discovered that the negative examples harder to classify are those with a high textural or structural content. Therefore, we build a difficulty criterion based on the high frequency content of the images. For each B_i , we extract a map that contains the edge response and we compute the number c_i of pixels whose edge response is above a threshold τ . Sorting the B_i s according to c_i permits to assign a difficulty score to the samples. During the learning phase, we adopt this ordering to feed the negative examples to the classifier (OPT 2). This permits to first construct simple decision boundaries, and then to build the more complex ones. This strategy decreases the risk of overtraining and improves the efficiency of both the learning and the detection phase, because the simplest negatives are filtered out very quickly.

C. More efficient analysis on \mathcal{M}

In the Riemannian manifold \mathcal{M} , there are three fundamental operations needed for boosting purposes. First, a mapping $\log_{\mu_l} : \mathcal{M} \mapsto \mathbb{R}^n$ called logarithmic map, that projects the input covariance matrices into the vector tangent space at a point $\mu_l \in \mathcal{M}$; in the tangent space, standard WLs like regressors or linear discriminants can be estimated. The second operation is the inverse mapping $\exp_{\mu_l} : \mathbb{R}^n \mapsto \mathcal{M}$, called exponential mapping, and the third is the centroid calculation, i.e. the operator that selects the projection point μ_l , that is the mean of an arbitrary set of points on \mathcal{M} .

The affine-invariant Riemannian framework of [5] deals with Sym^+ matrices. On one side, the log and exp mapping can be easily computed exploiting the Sym^+ matrices properties. On the other side, the calculus of centroid has no closed form, that makes it very slow.

Recently, novel metrics called Log-Euclidean are proposed in [1], that are similarity-invariant and have a closed form for the computation of the centroid μ_l . However, the computation of \log_{μ_l} and \exp_{μ_l} with these metrics is tricky and expensive, involving matrix differential calculations.

We propose to combine the two frameworks (OPT 3) gaining efficiency. We compute the log and exp operators in the affine-invariant way as in [5]. μ_l , instead, is calculated in the similarity-invariant way as:

$$\mu_l = \exp \left(\frac{1}{\sum_{i=1}^N w_i} \sum_{i=1}^N w_i \log(\mathbf{X}_i) \right), \quad (3)$$

where $\log(\mathbf{X}) = \mathbf{U} \log(\mathbf{D}) \mathbf{U}^T$ with $\mathbf{X} = \mathbf{U} \mathbf{D} \mathbf{U}^T$ is the eigenvalue decomposition of \mathbf{X} , and $\log(\mathbf{D})$ is the diagonal matrix composed by the eigenvalues' logarithms. We highlight the following equivalences:

$$\log(\mathbf{X}) = \log_{\mathbf{Id}}(\mathbf{X}), \quad \mathbf{X} \in Sym^+ \quad (4)$$

$$\exp(\mathbf{x}) = \exp_{\mathbf{Id}}(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n, \quad (5)$$

where $\mathbf{Id} \in Sym^+$ is the identity matrix. This fact comes from the corollary 3.7 in [1], where the equivalence of the tangent space at the identity matrix of Sym^+ and Sym is proved. This means that computing μ_l as in Eq. (3) implies to work on the Euclidean space of symmetric matrices Sym . We can map points from Sym^+ to Sym simply using the log operator. The result obtained on Sym is then mapped back to the Sym^+ domain with the exponential map. Thanks to this formulation, a centroid can be calculated approximately 20 times faster than using the formulation in [5].

D. More powerful WLs

We carefully analyze the type of WLs that form the boosting ensemble. In [7], the authors employ linear regression functions, suggesting that a further study on this aspect would be useful. In our analysis, after a preliminary study on several WLs, we select the polynomial functions (OPT 4). In fact, some weak learners, as for example linear regression functions, are unable to represent complex decision boundaries, while complex weak learners, as for example piecewise constant regression functions, quickly lead to overfitting. This class of WLs has several advantages: it is easily implementable, efficiently computable and flexible. In fact, weighted multidimensional polynomial fitting can be formalized as a linear problem [2]. A k -th degree polynomial in \mathbb{R} is defined as follows:

$$y = a_0 + a_1 x + \dots + a_k x^k, \quad (6)$$

where $y, a_0, \dots, a_k, x, \dots, x^k \in \mathbb{R}$. We can obtain the matrix form for the least-square fit by writing the Vandermonde matrix as a linear system:

$$\begin{bmatrix} 1 & x_1 & \dots & x_1^k \\ \vdots & x_2 & \ddots & x_2^k \\ 1 & x_n & \dots & x_n^k \end{bmatrix} \begin{bmatrix} a_0 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_k \end{bmatrix}. \quad (7)$$

Eq. (7) in matrix notation is $X \mathbf{a} = \mathbf{y}$, where each row in X represents an example of the training set. Generalizing the matrix form to a k -th order polynomial in \mathbb{R}^n , with no mixed terms, we can as well formalize the least square fit as a linear system:

$$[X_1 \dots X_N] \begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_N \end{bmatrix} = \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_N \end{bmatrix}. \quad (8)$$

After different trials, in our experiments we always consider second degree polynomials.

E. Occlusion modeling by WLS analysis

This feature (OPT 5) aims at refining a positive detection output, highlighting when and where the detected person is occluded by an object (the modeling of occlusions caused by people is faced in [7] already). This helps whenever the mere detection is followed by further analysis. In people re-identification, for example, the availability of genuine person’s details, minimizing the clutter, is very useful. The idea is to analyze the responses of the WLS, looking for possible agglomerations.

In detail, we test the presence of 4 different kinds of occlusions (see Fig.1a): TOP, BOTTOM, LEFT, RIGHT, parameterized by the size value s . The process exploits

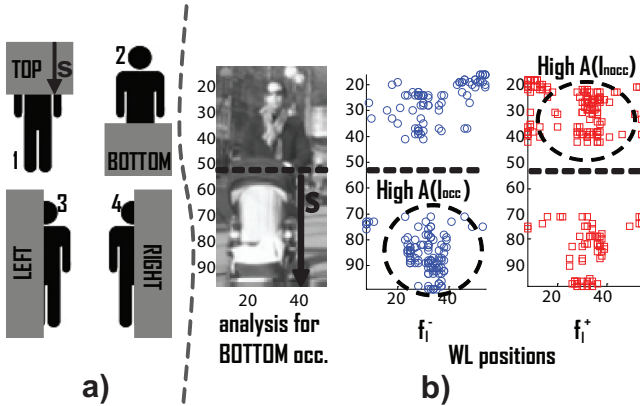


Figure 1. On the left, the kinds of occlusions modeled; on the right, WL responses for the image at the center.

the fact that each WL f_l focuses on a patch b_l , that is, a f_l judges a *portion* of the test image. Therefore, a compact, localized cluster of WLS, whose responses are positive, will indicate a human part, with high probability. On the contrary, a set of WLS with negative responses will probably indicate an occluding object. As visible in Fig.1b, each value of s determines a bipartition of the images, I_{occ} and I_{nocc} , where I_{occ} (I_{nocc}) is the occluded (not occluded) part. On I_{occ} , we compute the *partial agreement* $A(I_{occ}) = ||f_l^-|| / ||f_l^+ + f_l^-||$, that is the percentage of WLS in I_{occ} , not overlapping with the border instantiated by s (the dotted line in Fig. 1b), whose response is negative. A similar reasoning holds for $A(I_{nocc})$. We put together the two measures in $BS = A(I_{occ}) + A(I_{nocc})$. Maximizing BS over s for a single kind of occlusion gives s_{best} , i.e., the best occlusion’s size. Comparing the s_{best} s of each kind of occlusion gives the most probable occlusion. Experimentally, we set a threshold BS_τ below which BS does not represent an occlusion. This is reasonable, since the distribution of the WLS’ positive (negative) responses is uniform for a non occluded object.

III. EXPERIMENTAL RESULTS

We train our human detector on INRIA Person dataset, that contains 1774 images portraying humans, doubled though mirroring, and 1671 person-free images, all of size 64×128 . The setting for the training phase is the same of [7]. We implement both our approach and the original [7] with Matlab on an Intel 2.83 Ghz processor with 4 Gbytes of RAM. The training takes three days on average with our approach and more than two weeks with the original approach.

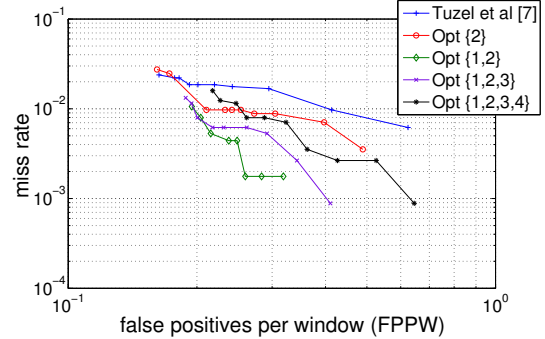


Figure 2. Comparison on the restricted dataset, adding one-by-one the OPTs.

First, we show the effects of each one of the proposed improvements wrt [7], on a randomly chosen subset of the INRIA dataset (500 positive and 1000 negative examples) in a cascade of 10 levels. We measure the performance by computing the Detection Error Tradeoff (DET) curve, that shows the tradeoff between true and false positives on a log-log scale. The results are on Fig. 2 (left). The y -axis corresponds to the miss rate $\text{FalseNeg}/(\text{FalseNeg} + \text{TruePos})$, and the x -axis corresponds to the false positives $\text{FalsePos}/(\text{FalsePos} + \text{TrueNeg})$, in this case the False Positives Per tested Window (FPPW). Please note that all the improvements OPT 1, 2, 3 and 4 provide an improvement in accuracy wrt [7].

Afterwards, we train a rejection cascade of 30 levels using the whole INRIA dataset, reproducing the system of [7]. The effects of our policy in selecting the examples for the cascade are evident in Fig. 3, that shows the number of weak learners per level. Our improvements produce a cut of the cascade complexity, evaluated as number of classifiers, of the 15% and 58%, using the linear and the polynomial regression, respectively. This obviously results in a faster learning and testing phase.

On the right of Fig. 4 we compare the framework in terms of DET curve. Both our detectors, using linear and polynomial regressors, have good generalization abilities, in a slightly different way. Indeed, we see that in the linear case at all the cascade levels, indicated by the markers, we achieve better performances in terms of miss rate, while we maintain the same performance of the original approach in terms of false positives. The polynomial case is instead close

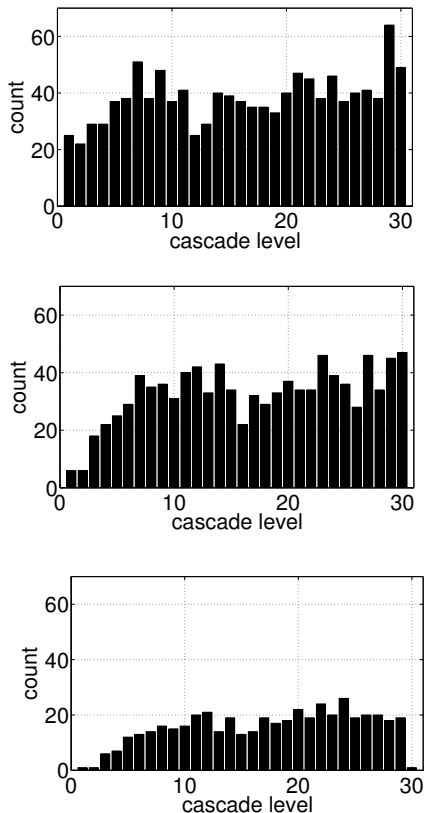


Figure 3. Comparison between cascades of 30 levels on the INRIA data set. Left: random selection of negative samples, as in [7]. Center: OPT 1,2,3 are applied. Right: all optimizations are exploited.

to the original approach in terms of miss rate, but it is the best in terms of false positives and the fastest.

Concerning the occlusion modeling (OPT 5), we evaluate qualitatively the occlusions of 200 positive detections, validating their correctness by subjective judgement (no ground-truth data is available). We reach an accuracy of 81%, where each occlusion detection can be correct ($= 1$) or not ($= 0$). Some results are shown in Fig.5.

IV. CONCLUSIONS

In this paper we show how to improve the state-of-the-art method [7] for pedestrian detection, allowing also an estimation of occlusions in a fine way. Our experiments show that we gain in efficiency and we set novel state-of-the-art accuracy results.

ACKNOWLEDGEMENT

This research is funded by the EU-Project FP7 SAMU-RAI, grant FP7-SEC-2007-01 No. 217899.

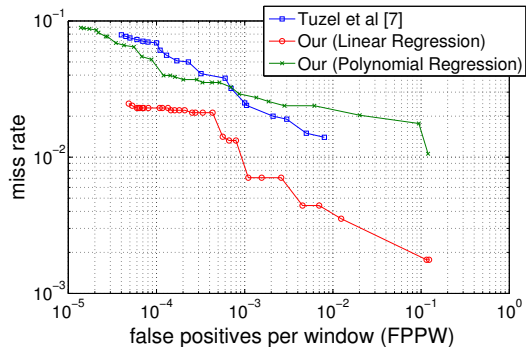


Figure 4. Comparison between [7] and our method (with and without OPT 4), on the complete dataset.

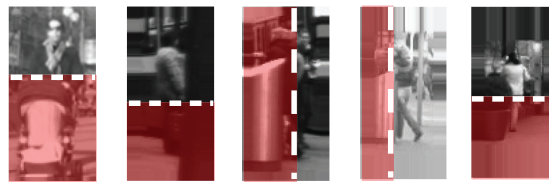


Figure 5. Five examples of occlusion modeling: in red the parts **detected** as occlusions.

REFERENCES

- [1] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 29(1):328, 2008.
- [2] T. Deng. Linear approach to the least-squares multidimensional polynomial fitting. In *Information, Communications and Signal Processing*, volume 3, 1997.
- [3] T. Kim and R. Cipolla. MCBoost: Multiple Classifier Boosting for Perceptual Co-clustering of Images and Visual Features. In *NIPS*, 2008.
- [4] R. Meir and G. Ratsch. An introduction to boosting and leveraging. *Lecture Notes in Computer Science*, 2600:118–183, 2003.
- [5] X. Pennec, P. Fillard, and N. Ayache. A Riemannian framework for tensor computing. *IJCV*, 66(1):41–66, 2006.
- [6] W. Schwartz, A. Kembhavi, D. Harwood, and L. Davis. Human detection using partial least squares analysis. In *ICCV*, 2009.
- [7] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on Riemannian manifolds. *IEEE PAMI*, pages 1713–1727, 2008.
- [8] B. Wu and R. Nevatia. Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses. *IJCV*, 82(2):185–204, 2009.