Multi-Stage Particle Windows for Fast and Accurate Object Detection

Giovanni Gualdi, Andrea Prati, *Member, IEEE,* and Rita Cucchiara, *Member, IEEE*

Abstract—The common paradigm employed for object detection is the sliding window (SW) search. This approach generates grid-distributed patches, at all possible positions and sizes, which are evaluated by a binary classifier: the trade-off between computational burden and detection accuracy is the real critical point of sliding windows; several methods have been proposed to speed up the search such as adding complementary features. We propose a paradigm that differs from any previous approach, since it casts object detection into a statistical-based search using a Monte Carlo sampling for estimating the likelihood density function with Gaussian kernels. The estimation relies on a multi-stage strategy where the proposal distribution is progressively refined by taking into account the feedback of the classifiers. The method can be easily plugged in a Bayesian-recursive framework to exploit the temporal coherency of the target objects in videos. Several tests on pedestrian and face detection, both on images and videos, with different types of classifiers (cascade of boosted classifiers, soft cascades and SVM) and features (covariance matrices, Haar-like features, integral channel features and histogram of oriented gradients) demonstrate that the proposed method provides higher detection rates and accuracy as well as a lower computational burden w.r.t. sliding window detection.

Index Terms—Efficient object detection, pedestrian detection, coarse-to-fine search refinement.

1 Introduction

Object detection and recognition are two foundational problems in computer vision. Object detection can be viewed also as a classification problem in a two-class case (object and non-object), that without segmentation is carried out over suitable feature vectors extracted from image patches (or windows). Many classifiers are based on *bagging* and *boosting* [1], [2], [3], [4] with a good trade-off between accuracy and efficiency.

According to Enzweiler and Gavrila [5] the typical components of detection-without-segmentation in images are *hypothesis generation* (or ROI selection) and *hypothesis validation* (or classification); in the case of videos a third component, i.e. *hypothesis tracking* must be accounted. The hypothesis generation requires an effective and efficient search of the best locations and scales to look at; then the hypothesis validation component verifies whether the target object is present or not.

This paper addresses the problem of an effective and efficient ROI selection for performing hypothesis generation. Regarding this matter, many perception psychologists, starting from Treismann in early '80 [6], proposed the dichotomy between the "parallel" and the "serial" search. They stated that the target object, if clearly distinguishable, can be detected in a parallel way, employing a time approximately independent on the data dimension. Instead, in the presence of multiple targets, many "distractors", or complex patterns, our perceptual behavior becomes sequential, based on a serial search, so that the recognition time becomes proportional to the data dimension.

Coherently, in computer vision, whenever the shape becomes more complex and many targets and distractors are present, detection without segmentation normally exploits the *"sliding window"* approach (SW hereinafter) and spans the search over the whole image in a serial fashion (e.g. [7], [8], [9]). All the possible patches are extracted. Then, the selected features are computed on the patch and the obtained feature vector is passed to a binary classifier. If the object can have different sizes, also the scale is to be included as a state variable.

Let N_{SW} be the number of windows to be evaluated. N_{SW} is a function of the image size, of the sliding steps (Δ_x, Δ_y) (also known as *pixel strides*) and of the scale search, that is regulated through a scale step Δ_s (also known as *scale stride*). Δ_x, Δ_y and Δ_s depend on the classifier and on the application context: typical values are in the order of few pixels (4-6-8) for Δ_x, Δ_y and between 1.05 and 1.2 for Δ_s. Given these premises, N_{SW} is in the order of several tens or even hundreds of thousands. The problem of SW approach is thus twofold: on the one hand, the computational time is proportional to N_{SW}, that should be kept limited because of time constraints; on the other hand, the detection rate and localization accuracy decrease when the pixel and scale stride increase. As a consequence, a trade-off between efficiency and accuracy is unavoidable.

One way to handle the huge number of windows to be analyzed in an efficient manner is the use of a *cascade of classifiers*, that is a degenerated decision tree where, at each stage (or *layer*) of the cascade, a classifier is trained to detect almost all target objects while rejecting a certain fraction of the non-object instances [10]. The key strength of cascades is the capability to quickly reject

• *G. Gualdi and R. Cucchiara are with the D.I.I., while A. Prati is with D.I.S.M.I., both of University of Modena and Reggio Emilia.*

Digital Object Indentifier 10.1109/TPAMI.2011.247

most of the true negatives in the early layers of the cascades using simpler classifiers, while retaining more complex classifiers in the later layers where possibly a very minor portion of windows remain to be analyzed. This is particularly convenient whenever the a priori probability of occurrence of non-objects is much greater than that of objects, that is also called *rare event detection problem*, as in the case of detecting faces in an image [11].

For this reason, many proposals use a *cascade of boosted classifiers* (or *cascade of boosted ensembles* [12]). In particular, boosting algorithms consist in the combination of several weak hypotheses (or weak learners) to build more accurate hypotheses called *strong classifiers*.

Our work proposes a new search mechanism that overcomes the serial search of sliding windows by using a data-driven and focused search. We will assume that the search can start in parallel, uniformly over the image, in accordance with the theory of attentive vision and the saccade search [13], [14], [15] and we focus, in an iterative manner, on the exploration of the image toward the area where the target objects are more likely to be found, as a model of *coarse-to-fine* detection [16], [17], [18], [19]. The paradigm here proposed, called *multi-stage particle-window (MS-PW)*, provides an incremental estimation of a likelihood function through Monte Carlo sampling, exploiting the confidence (or *response*) of the classifier only, i.e. without other orthogonal features like motion, perspective, depth, etc. In practice, this response is employed to increasingly draw samples on the areas where the objects are potentially present and avoiding to waste search time over other regions. We call these samples *"particle windows"* (PWs) in juxtaposition to the uniformly grid-distributed *sliding windows* (SWs).

This approach for window selection using particles is complementary with other optimizations which aims at reducing the number of hypotheses and it is basically *classifier-independent*, i.e. it works with any classifier which can have an associated confidence measure. Indeed, in this paper we will report the results on four different classifiers, used for pedestrian or face detection: two are cascades of boosted classifiers (covariance-matrix-based pedestrian detector [9], here called CovM, and Haar-based face detector [10], called HLF), one is a monolithic classifier (HoG-SVM pedestrian detector [8], called HOG) and one is a sort of single layer cascade called soft cascade ("The Fastest Pedestrian Detector of the West" [20] based on Integral Channel Features [21]).

Typically, classifiers have a degree of robustness to translation and scaling, meaning that the confidence measure provides high responses not only in exact correspondence of a true positive, but also in its close neighborhood, whose size depends on the classifier, and degrades monotonically when moving further away. This behavior generates a "basin of attraction" around true positives, both in location and in scale, that is exploited by our proposed method for converging in the area where objects are present. In this paper we will give evidence of the basin of attraction in all the four classifiers used. It typically happens that also false positives create a basin of attraction for the classifier. However, even if the underlying model of the basin of attraction has similar shape on both cases, it differs in the score: indeed true positives in general reach higher response values than false positives. The multi-stage Monte-Carlo sampling employed in our method is designed to be incrementally lead by measurements (i.e. classifier responses), therefore it is eventually attracted prominently by true positives rather than false positives.

A second contribution of our work is that the above approach can be easily plugged in a Bayesian-recursive filter for videos. Although this technique is often exploited for object tracking [22], [23], [24], our proposal does not aim to that achievement; rather we use a recursive framework to exploit the temporal coherency of the objects in order to further increase efficiency and accuracy of object detection also in very cluttered scenes. In addition, our proposal is capable to handle a variable number of objects without additional computation burden, thanks to a *quasi-random sampling* procedure and a measurement model (i.e. the measurement obtained through the classifier) that is exactly the same for all target objects.

The experiments, carried on using very different setups and targeting both pedestrian and face detections, demonstrate that the MS-PW is definitely convenient on SW, whichever classifier is used: configuring the two approaches to have similar detection accuracy, MS-PW is more efficient; conversely, configuring them to have similar computational time, MS-PW shows better detection and localization accuracy.

The paper is structured as follows: after the related works, divided in Cascades of Boosted Classifiers (Sect. 2.1) and Solutions of ROI Selection (Sect. 2.2), we report the proposed response measure applied to the classifiers (Sect. 3.1) and describe the multi-stage particle window approach on single images (Sect. 3.2) and on videos (Sect. 3.3). Experimental results are divided in the description of the classifiers used (Section 4.1), the evaluation benchmark (Sect. 4.2), the algorithm parameters (Sect. 4.3), and eventually of the results on images (Sect. 4.4) and videos (Sect. 4.5).

2 Related Works

This paper proposes a new approach for hypothesis generation which has been conceived for cascades of classifiers for object detection. Thus, this section overviews both the current popular classifiers, focusing on their use of sliding window search, and the related works on hypothesis generation.

2.1 Cascades of Boosted Classifiers

Boosting is very popular for window classification, initially proposed by Freund and Schapire in 1996 [2]. Different types of weak and strong classifiers have been studied; examples of weak learners are simple binary

Ref.	Object	feature	weak cl.	strong cl.	# layers	# weak cl.	training set	avg. # windows
[25] (2003)	Face	enh. HLF	stumps or CART	AdaBoost/ Gentle Ad.	10-20	520-900*	MIT+CMU[1] (PT=5000, NT=3000)	≃426K* (ps=1,ss=1.1) ≃243K* (ps=1,ss=1.2)
[10] (2004)	Face	HLF	single perceptron	AdaBoost	32	4297	MIT+CMU[1] (PT=4916, NT=10000)	≃145K* (ps=1.5,ss=1.25)
[26] (2004)	Face	mLBP	single perceptron	AdaBoost	4	≃1000	MIT+CMU[1],BioID[1] (PT=6000, NT=2000)	≃197K* (ps=1,ss=1.25)
[27] (2005)	Face	HLF	single perc.+FDA	AdaBoost/ AsymmB.	21-22	16233	MIT+CMU[1] (PT=5000, NT=5000)	≃197K* (ps=1,ss=1.25)
[28] (2006)	Hand gesture	same as [25] + Double L	single perceptron	AdaBoost	avg. 14	avg. 123	own dataset (PT=1000, NT=3476)	≃320K* (ps=1*,ss=1.1)
[29] (2006)	Pedes.	variable size HOG	linear SVM	AdaBoost	30	≃138	INRIA[1] (PT=2418)	12800 (ps=8*,ss=1.25*)
[11] (2008)	Face and eyes	HLF and and mLBP	stumps	AdaBoost	10	≃970	MIT-CMU[1], others, own (PT=5000, NT=3600, PV=flip PT, NV=1500)	≃230K* full search (ps=6-3-1,ss=1.2)
[30] (2008)	Pedes.	LRF,HOG, COV.MAT.	binary decision	AdaBoost/ SVM	20-29	1000-2000	INRIA[1], [31] (PT=2418,4800, NT=UNK,5000)	17280 (ps=4,ss=1.25)
[9] (2008)	Pedes.	Cov.Mat.	Linear logistic regressor	LogitBoost	30	≃580	INRIA[1], [31] (PT=2418,4800, NT=10000,5000)	≃28K* (ps=6,ss=1.2)
[32] (2009)	Face	SRF	single perceptron	AdaBoost	17	1422	MIT+CMU[1],others (PT=4858, PV=511)	≃550K* (ps=1,ss=1.25)
[33] (2010)	Car	HLF+HOG	single perceptron	AdaBoost/ SVM	32	4297	MIT CBCL[1] (PT=516, NT=500)	≃25K* (ps=1.5,ss=1.25)
[20] (2010)	Pedes.	ICF	two decision tree	AdaB/RealB /LogitB	1	≃1000	INRIA[1], [34], TUD-Brussel[1] (PT=2418,192K,400, NT=15000)	≃145K (ps=4,ss=1.07)

TABLE 1: Summary of proposals for object detection with cascades of boosted classifiers (*=deduced value, PT=positive training, NT=negative training, PV=positive validation, NV=negative validation, ps=pixel stride, ss=scale stride; HLF=Haar-like features [10], CART=Classification and Regression Tree [35], mLBP=modified Local Binary Patterns [36], HOG = Histogram of Oriented Gradients [8], SRF=Scattered Rectangular Features [32], FDA=Fisher Discriminant Analysis [37], LRF=Local Receptive Fields [38], COV.MAT.=Covariance Matrix [9], ICF=Integral Channel Features [21])

decisions [30], stumps [11], [25], single perceptrons [10], semi-naive Bayes classifiers [39] or linear logistic regressors [9]; among the different algorithms for strong classifiers, AdaBoost is probably the most famous, being used in the Viola-Jones face detector [10]. Lienhart *et al*.in [25] compared the performances of Discrete AdaBoost, Real or Gentle AdaBoost [2] in face detection, resulting in a slight preference for Gentle AdaBoost. Ong and Bowden in [40] proposed the FloatBoost algorithm [41] which adds an additional step to AdaBoost to remove the excessive weak classifiers that no longer contribute to the detection. Wu *et al*.in [27] presented an effective closed form approximation of the optimal solution, called Linear Asymmetric Classifier (LAC) based on AsymmBoost [42].

The use of cascade of boosted classifiers for object detection is very broad. Table 1 summarizes several recent works reporting: the type of target object; the chosen features, the weak and strong classifiers; the number of layers in the cascade and the total number of weak classifiers; the training sets used and, most important, the number of windows analyzed at detection time: when this number is not explicitly provided, it is estimated applying the declared pixel and scale strides, reported in brackets, to the test dataset. The seminal work has been proposed by Viola and Jones on face detection [10], which introduced three novelties: *integral images* for speeding up the computation in the case of rectangular non-oriented features; an efficient learning algorithm for AdaBoost on Haar-like features (HLF); the cascade of AdaBoost classifiers for fast and accurate face detection. Lienhart *et al.* [25] proposed an enhanced version with Haar features rotated by 45 degrees.

HLF has been further modified in [32] where a new feature called *scattered rectangle features* (SRF) is proposed. SRF is a Haar-like feature which includes 2-rectangle features only (differently from original HLF that has 3- and 4-rectangle features) and allows them to be misaligned, overlapped or even detached, which produces an over-complete set of features (more than 1.5 million on a 19x19 patch, w.r.t. about 64K in the case of HLF) capable to better represent the object/face. The authors of [26] proposed a method for face detection based on the modified Local Binary Patterns (mLBP) or census transform, that is invariant to local illumination changes. The work in [11] provides a comparison between HLF and mLBP in detecting faces and eyes.

The Viola-Jones approach has been borrowed also for

1. MIT+CMU: http://vasc.ri.cmu.edu/idb/html/face/frontal_images; UIUC: http://l2r.cs.uiuc.edu/~cogcomp/Data/Car; BioID: http://www.bioid.com/support/downloads/software/bioid-face-database.html; INRIA: http://pascal.inrialpes.fr/data/human; VOC2006: http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2006; MIT CBCL: http://cbcl.mit.edu/cbcl/software-datasets/CarData.html; TUD-Brussels: http://www.mis.tu-darmstadt.de/tud-brussels.

detecting other types of objects: in [43] for humanoid robots and in [28] for 14 different hand gestures with the addition of a "Double L" feature to the standard HLF; Ong and Bowden in [40] exploit a similar approach for hand shape recognition using the already-cited FloatBoost as strong classifier. In addition, the *Integral Channel Features* (ICF) [21] apply linear and non-linear filters on the image and then efficiently compute simple features such as local sum, histograms and HLF, using integral images. ICF have been successfully applied to pedestrian detection exploiting gradients and their histograms, and LUV color channel.

When the objects to be detected are pedestrians, HLF has proved to be not much accurate [7] and other features are more common. The *Histograms of Oriented Gradients* (HOG) [8] are certainly very widespread. HOGs are often coupled with a monolithic classifier, such as linear SVM, or embedded in a coarse-to-fine hierarchy of features [16], [18], but are also employed with cascades of classifiers. For example, Felzenszwalb *et al.*in [44] have recently proposed a very interesting approach that speeds up detection using part-based models and HOG features through cascade classifiers (though not boosted); actually, this approach is suitable for detection of several classes of objects. In [29] a cascade of 30 AdaBoost classifiers composed of linear SVMs as weak learners is proposed for pedestrian detection. The size of the HOG patches is varied in order to allow fast rejection of non-objects in early layers.

Another effective feature for pedestrian detection described by Tuzel *et al.* [9] is the *covariance matrix* (see Section 4.1.1). To account for the high-dimensional nature of this feature which lies on Riemannian manifolds, Tuzel *et al.*proposed linear logistic regressors as weak classifiers and, as a consequence, LogitBoost [45] as strong classifier. Also the paper in [30] adopts covariance matrix, but the authors work on Euclidean spaces thanks to a weighted Linear Discriminant Analysis (wLDA). In [9] authors showed that covariance matrices outperform HOGs for pedestrian detection.

Regardless of which features, weak and strong classifiers are employed, a valuable effort has been spent also in changing the architecture of the cascade. In [39] the authors define a faster approach which moves the paradigm from *window-centric* to *feature-centric*: in practice, the first layer of the cascade re-uses feature evaluations among overlapping windows, whereas in window-centric approaches all the evaluations are with respect to a single classification window. Feature-centric evaluation resulted to be more efficient in the early layers where the goal is to quickly remove non-object windows, but does not change the number of windows to be evaluated. Another very interesting architectural variant is the so-called *nested cascade* [11], [12], [43], [46], which makes use of the technique known in machine learning with the term *recycling*: in other words, the confidence of layer i is reused by layer $i+1$ to obtain higher classification accuracy.

Since the proper use of cascades of boosted classifiers for object detection is still debated, there are also several works surveying existing approaches and comparing the different choices. For instance, Wojek and Schiele in [47] give a nice overview of possible features and classifiers for pedestrian detection and how they are usually combined. They take into consideration Haar wavelets, HLF, HOG, shapelets [48] and shape contexts as features, and linear SVM, RBF (Radial Base Function) SVM, and AdaBoost as classifiers. Moreover, a new feature called *dense shape contexts* is proposed. The combination of multiple features, i.e. Haar wavelets with HOG and dense shape contexts, is shown to bring some benefits in terms of accuracy. The survey in [31] compares different features (PCA coefficients, LRF, HLF and Haar wavelets), different classifiers (feed-forward neural network, polynomial SVM, RBF SVM and K-nearest neighbors) and different approaches for increasing the negative training size (bootstrapping or use of cascades). All these combinations are tested against the Viola-Jones approach presented in [7]. The authors concluded that global features, such as PCA coefficients, are less performing than local features, and that among the latter, adaptive features (such as LRF) are the most promising. Regarding the classifiers, SVMs and AdaBoost resulted to be the most accurate. On the same path, the survey in [5] compares a wide range of approaches (HLF+AdaBoost [7], LRF+neural network [38], HOG+linear SVM [8] and a combined shape-texture detection [49]) on several datasets with the specific goal of pedestrian detection for vehicle applications, thus requiring fast detections. The HOG-based approach shows the best performance when no constraints on time are posed (it takes about 2.5 seconds per image), while HLF-based approach outperforms others when real-time performances are required (about 250 ms per image).

The approach that we are proposing is applicable to any of the methods described so far.

2.2 Solutions for ROI Selection

One crucial consideration for our work is that all the approaches reported above make use of a SW search, which has the drawback of brute force methods, that is the high computational load due to the number of windows to check: see for instance, the high average number of windows in Table 1, rightmost column, where most of the approaches evaluate hundreds of thousand of windows. Consequently, several works focus on the reduction of the computational burden, following three main streams: (a) pruning the set of sliding windows by exploiting other cues (e.g. motion [50], depth [51], geometry and perspective [52], [53], or whatever cue that is different from the appearance cue specific of the detector itself); (b) speeding up with hardware-optimized implementations (such as GPUs [54]); (c) efficiently exploring the sub-window space through optimal solution algorithms [11], [15], [16], [18], [19], [39], [55].

Our proposed method belongs to the latter class, that we examine here in slightly higher detail: in [55], the authors propose to bypass SW through an efficient subwindow search using a branch and bound technique. However this method have strict requirements over the "quality function" (i.e., classifier score) that are not met by most of the aforementioned classifiers; additionally it detects only one object at a time (i.e., it finds the global maximum of the function), requiring multiple runs to detect multiple objects; in [11] the comparison between a SW approach (called "full search" by the authors) and a more efficient solution (called "speed search") is proposed. The "speed search" employs a multi-grid approach for selecting the windows' positions to be evaluated. At the first level, a pixel stride of 6 is employed: if the confidence of the classifier is above a threshold, a second level with pixel stride 3 is analyzed. The windows resulting in a classifier's confidence higher than a second threshold are further refined by fixing pixel stride equal to 1 in a 3x3 neighborhood. This method is faster than SW detection but suffers in detection accuracy in most of the tested datasets.

Another deterministic coarse-to-fine refinement has been proposed in [16], with a deterministic (grid-distributed), multi-stage (coarse-to-fine) detection: successful detections at coarse resolutions yield to refined searches at finer resolutions. Also in [18] a similar approach is proposed, using a deterministic multiple-resolution search with grid distributed scanning. This approach adds (loose) prior spatial constraints on the object locations based on the radius of the neighborhood. A too small radius will result in a small speedup improvement, while a large radius will most likely miss object which are close each other (only a single hit can be found on the coarse grid in level 0 of their approach). Our statistical approach does not employ a rigid grid structure in the refinement of the search, allowing the "radius" to be adapted based on the response of the classifier. Butko and Movellan in [15] explore the maximization of information gain through a computational approach that simulates a digital fovea. Although it obtains speed-ups that are comparable to ours, two limitations are suffered: a slight degradation of performances w.r.t. sliding window detection (instead we obtain higher accuracy, as shown in Section 4) and single-target detection (conversely our method is intrinsically multi-target). A particular case is reported in [19] where a coarse-to-fine approach is used in the context of face detection. The coarse-to-fine strategy is used both in the exploration of face poses (thus reducing the space exploration time, similarly to our proposal as concept) and in the feature representation of faces (thus sharing some properties of cascading of classifiers - including Viola-Jones face detector - where conjunctions of tests of increasing complexity are used to quickly identify negatives - background - in the image). Most of these mentioned approaches binarize (through the use of thresholds) the response of the classifier at each stage, while we propose to exploit its continuity, in order to be able to find true detections even when at earlier stages no successful detections are found.

3 Multi-Stage Particle Windows Approach

When the classifiers are applied to detection through the SW approach, they incur in the two-fold problem of a large waste in computational time searching over areas where objects are not present and the need of a massive Sliding Window Set (SWS, the set of all the windows to be analyzed by the SW detection) to find every object in the scene. Within the SW approach, a window w is typically defined by the 3-dimensional vector (w_x, w_y, w_s), being, respectively, coordinates of the window center and window scale w.r.t. a given size; aspect ratio and rotations are usually discarded.

The objective of the proposed paradigm, called *multi-stage particle-window (MS-PW)*, is to provide a non-uniform quantization over the state space and to model the detection as an estimation of the states given the observations; we aim at estimating the modes of the probability density function $p(\boldsymbol{X}|\boldsymbol{Z})$, where $\boldsymbol{X} = (w_x, w_y, w_s)$ is the state and $\boldsymbol{Z}$ corresponds to the image.

3.1 Detection Response of the Classifiers

In the case of cascade classifiers, every window $w \in SWS$ is passed to the cascade classifier $\mathbf{C}$, where each layer C_i responds with either object or non-object (i.e., $class(w, C_i) = \{\text{object,non-object}\}$). The final classification result $class(w, \mathbf{C})$ is obtained as:

$$class(w, \mathbf{C}) = \begin{cases} \text{object} & \text{if } class(w, C_i) = \text{object}, \forall i = 1, \ldots, L \\ \text{non-object} & \text{if } \exists j \leq L | class(w, C_j) = \text{non-object} \end{cases} \quad (1)$$

where L indicates the number of layers in the cascade. In this case we introduce the *"detection response"* $R(w)$ as

$$R(w) = \frac{j_w}{L} \quad (2)$$

where j_w is the index j of the last cascade which provides a positive classification for w and $R \in [0, 1]$. Given the structure of rejection cascades, the higher the degree of response $R(w)$ is, the further w reached the end of the cascade, the more similar it is to the object model, up to the extreme of $R(w) = 1$, that means successful classification with $j_w = L$.

Eq. 2 holds also in case of soft cascades (as in [21]), where the classification {object,non-object} is evaluated at each weak learner: applying the same notation as above to this type of classifiers, C_i is a weak learner, with $i = 1, \ldots, L$, with L being the number of weak learners.

Non-cascade classifiers like SVM (described together with HOG features in section 4.1.4), typically provide a "margin" M (i.e., the distance to the class-dividing boundary), which extends over $\Re$ or $\Re^+$: in such cases, we obtain a detection response R consistent to the one proposed for cascade classifiers applying an appropriate

function that translates M to the range $[0,1]$. To this aim, hard clipping or soft clipping functions can be used [8]; we propose the use of the latter, through the implementation of a sigmoid function that provides a smooth transition across the margins:

$$R(w) = \frac{1}{1+\exp\left(a\left(M(w)+c\right)\right)} \quad \text{with} \quad M \in (-\infty, +\infty) \quad (3)$$

The two function parameters a and c can be learned using the Platt algorithm [8] from a training set; the only constraint is $a \in (-\infty, 0)$, to make the sigmoid function monotonically increasing, i.e. measuring similarity.

Let's recall that the cardinality of the SWS depends also on the degree of coarseness for the scattering of the windows, i.e. pixel and scale strides. For a successful detection, the SWS must be rich enough (i.e. the strides must be small enough) so that at least one window hits each target object in the image. Actually, every classifier has a degree of non-sensitivity to small translations and scale variations: in other words, the evaluation of the classifier in the close neighborhood (both in position and scale) of the window encompassing a target, remains positive ("region of support" of a positive detection). Having a sufficiently wide region of support allows to uniformly prune the SWS, up to the point of having at least one window targeting the region of support of each target in the frame. Vice versa, a too wide region of support could generate de-localized detections [55]. The detection response R(w) proposed in eq. 2 and 3 effectively models the regions of support and the basins of attraction of the classifiers (evidence of this over four different types of classifiers is provided in section 4.1) and can be used for supporting a more efficient hypothesis generation.

3.2 Multi-Stage Kernel-based Density Estimation on a Single Image

Let us not consider any a prior information in the image in order to provide a general solution. Consequently, the state pdf can be assumed proportional to the measurement likelihood function, i.e. $p(\boldsymbol{X}|\boldsymbol{Z}) \propto p(\boldsymbol{Z}|\boldsymbol{X})$. The extension to the case where $p(\boldsymbol{X}|\boldsymbol{Z}) \propto p(\boldsymbol{Z}|\boldsymbol{X}) \cdot p(\boldsymbol{X})$ is straightforward by plugging into the procedure specific strategies for window selection that represent the priori, for instance based on motion, geometry, depth, etc..

The likelihood function is estimated by an iterative refinement through m stages based on the observations. Algorithm 1 shows the complete procedure. Without any prior information, the initial proposal distribution $q_0(\boldsymbol{X})$ is set to a uniform distribution on the state space and it is sampled for extracting the first set S_1, containing N_1 samples or "particle windows" (pw) (see lines 1 through 6 of Algorithm 1). Each particle represents a window $w = (w_x, w_y, w_s)$. Scattering particles according to a uniform distribution is somehow similar to the sliding window strategy. However, in our case the particles are equally distributed only from a statistical point of view and are not deterministically defined; nonetheless, the

Algorithm 1 Measurement Step

1: Set $q_0(\boldsymbol{X}) = U(\boldsymbol{X})$
2: Set $S = \emptyset$
3: **for** $i = 1$ **to** m **do**
4: **begin**
5: Draw N_i particle windows from $q_{i-1}(\boldsymbol{X})$:
6: $S_i = \left\{ pw_i^{(j)} | pw_i^{(j)} \sim q_{i-1}(\boldsymbol{X}),\ \ j = 1, \ldots, N_i \right\}$
7: Assign a Gaussian kernel to each particle window:
8: $\mu_i^{(j)} = pw_i^{(j)} \quad ; \quad \Sigma_i^{(j)} = \Sigma_i$
9: Compute the measurement on each particle window $pw_i^{(j)}$:
10: $l_i^{(j)} = R^{\lambda_i}\left(\mu_i^{(j)}\right)$ with $R^{\lambda_i} \in [0,1]$
11: Obtain the measurement density function at step i:
12: $p_i(\boldsymbol{Z}|\boldsymbol{X}) = \sum_{\pi_i^{(j)} \neq 0} \pi_i^{(j)} \cdot \mathcal{N}\left(\mu_i^{(j)}, \Sigma_i^{(j)}\right)$
13: where: $\pi_i^{(j)} = \frac{l_i^{(j)}}{\sum_{k=1}^{N_i} l_i^{(k)}}$
14: Compute the new proposal distribution:
15: $q_i(\boldsymbol{X}) = (1-\alpha_i)\, q_{i-1}(\boldsymbol{X}) + \alpha_i \frac{p_i(\boldsymbol{Z}|\boldsymbol{X})}{\int p_i(\boldsymbol{Z}|\boldsymbol{X}) d\boldsymbol{X}}$
16: Retain only the particle windows with measurement value 1:
17: $\widetilde{S}_i = \left\{ pw_i^{(j)} \in S_i | R\left(\mu_i^{(j)}\right) = 1,\ \ j = 1, \ldots, N_i \right\}$
18: $S = S \bigcup \widetilde{S}_i$
19: **end**
20: Assign Gaussian Kernel $\overline{\Sigma}$ to each $pw \in S$
21: Run the Sequential Kernel Density Approximation [56], and obtain a Mixture of Gaussians $\mathcal{M}$
22: $\mathcal{M}$ is the final likelihood function $p(\boldsymbol{Z}|\boldsymbol{X})$
23: Each mode of $\mathcal{M}$ represents an object detection

N_1 particles could also be grid-distributed as in SW and this would not affect the bottom line of the proposed method. The key point here is N_1 being one order of magnitude lower than the cardinality of a typical SWS. The rationale is that part of these uniformly distributed particles will fall in the basins of attraction of the target objects in the image and will be used to provide an initial rough estimation of the measurement function. Being driven by the measurements, at any stage i, the distribution q_i is progressively refined and then sampled; this procedure produces a growing confidence over the proposal and makes it possible to decrease, from stage to stage, the number of N_i to sample (as visually depicted in Fig. 1). The final aim is that the total number of particle windows $N_{PW} = \sum_{i=1}^{m} N_i$ be definitely lower than the fixed number of windows N_{SW} of the SWS.

The N_1 samples drawn from $q_0(\boldsymbol{X})$ (line 6) provide a first approximation of the measurement density function p_1, through a Kernel Density Estimation (KDE) approach with Gaussian kernel, generating a mixture of N_1 Gaussians: for each j-th component, mean, covariance and weight are defined. The mean $\mu_i^{(j)}$ is set to the j-th particle window value $pw_i^{(j)} = \left(w_{x,i}^{(j)}, w_{y,i}^{(j)}, w_{s,i}^{(j)}\right)$; the covariance matrix $\Sigma_i^{(j)}$ is set to a covariance Σ_i (line 8), which, at any given stage i, is constant for all particle windows. The choice of Σ_i is worth a more in-depth discussion. For instance, the authors in [23] that use a similar method for object tracking, proposed to determine the Σ of each sample as a function of its k-nearest neighbors; this strategy yielded fairly unstable covariance estimations when

Fig. 1: Distribution of particle windows across stages in MS-PW applied to pedestrian detection. The number of stages in this example is $m = 5$ and the samples are $(2000, 1288, 829, 534, 349) = 5000$. Circles in (f) represent the particle windows that are retained in S (line 18 of Alg. 1), i.e. the set of particles that triggered a successful classification.

applied to our context: indeed, given the low number of samples used in our method, k should be kept fairly low to maintain a significance over the covariance estimation: this makes the estimation quite dependent on the specific randomized sample extraction. We preferred to assign an initial Σ_1 proportional to the size of the region of support of the classifier, and to decrease the Σ_i of the following stages: this has the effect of incrementally narrowing the samples scattering, obtaining a more and more focused search over the state space.

Finally, the response $R(pw)$ of the specific classifier (see section 3.1) to the particle window pw is exploited to determine the weight $\pi_i^{(j)}$ of the j-th Gaussian component. The intention is that those particle windows falling on a basin of attraction, i.e., close to the mode/peak of the distribution to estimate, shall receive higher weights with respect to the others, so that the proposal distribution q_i, that is partly determined by p_i, will drive the sampling of the next stage more toward portions of the state space where the classifier yielded high responses. Conversely, sampling must not be wasted over areas with low response of the classifier. In other words, these weights must act as attractors which guide the particle windows toward the peaks. This is accomplished by connecting the weights $\pi_i^{(j)}$ to the response R of the classifier in the sample location $\mu_i^{(j)}$ (line 10). The exponent λ_i is positive and increases at every stage: at early stages, $\lambda_i \in (0; 1)$, therefore the response of the samples is quite flattened, in order to treat fairly equally the whole range of not null responses, granting similar trust to medium and highly responsive particles; at later stages λ_i grows beyond 1, so that only the best responses will be held in account, while the others will be inhibited. In section 4.3 we will also propose a simplified setup of these parameters.

The behavior of the algorithm is clearly shown in Fig. 1, where the samples at subsequent stages decrease in number but concentrate more and more on the peaks of the distribution, i.e. where the response of the classifier is higher.

The measurement density function $p_i(\boldsymbol{Z}|\boldsymbol{X})$ (line 12) represents a partial estimation of the likelihood function through a Gaussian kernel density estimation. This function is linearly combined with the previous proposal distribution $q_{i-1}(\boldsymbol{X})$ to obtain the new proposal distribution (line 15), where α_i is an *adaptation rate*. The process is iterated for m stages and at the end of each stage only the particle windows pw of S_i that triggered a successful object detection are retained (line 17) and added to the final set of particle windows S (line 18). The number m of iterations can be fixed [23] or adjusted according to a suitable convergence metric, like the entropy, intended as a measure of uncertainty of a continuous density [57]. Eventually, recalling that the number of particles to sample decreases from stage to stage, the whole process could be interrupted when the number of particles to sample is lower than a defined bound.

The above procedure allows multiple detections for each target object, possibly at different scales and positions, and a *non-maximal suppression* step is necessary for determining the right scales and positions of the final detections. This problem is common to object detection and many heuristics have been adopted: Viola and Jones in [10] proposed a very simple fusion strategy which includes in the same subset all the detections with overlapped bounding regions. In [11] the *detection volume* is defined as the sum of all the confidence values corresponding to overlapped windows: if the detection volume is greater than a threshold and the number of overlapped windows is sufficient, then the detections are fused. Other works, like [9], [58], employ the mean-shift algorithm.

Since we have followed a statistical approach, we propose to envision the non-maximal suppression as mode seeking over the likelihood density function. In particular, we run the sequential kernel density approximation as proposed in [56] over S, that is the set of samples that yielded successful detections. This method approximates an underlying pdf represented by a set of samples (S in our case), with a mixture of Gaussians in a time that is linear with the cardinality of the sample set. Through this fast procedure we obtain a two-fold result (lines 21-23): first, we get the non-maximal suppression for object detection, since the detected objects correspond to the means of each single Gaussian components of the resulting mixture; second, we get a very compact representation of the likelihood density function $p(\boldsymbol{Z}|\boldsymbol{X})$, that can be easily plugged inside a Bayesian recursive filter, as shown in the next Section.

3.3 Kernel-based Bayesian Filtering on Videos

We extend here the previous method to the context of videos, by propagating the likelihood in a Bayesian-recursive filter. Differently from tracking approaches, the conditional density among frames (observations in time) is not used here to solve *data association*. Instead, the recursive nature of Bayesian filtering exploits temporal coherence of objects only to obtain a further improvement over detection. In the sequential Bayesian filtering framework, the conditional density of the state variable given the measurements is propagated through prediction and update as:

$$p\left(\boldsymbol{X}_t|\boldsymbol{Z}_{1:t-1}\right) = \int p\left(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}\right) p\left(\boldsymbol{X}_{t-1}|\boldsymbol{Z}_{1:t-1}\right) d\boldsymbol{X}_{t-1} \quad (4)$$

$$p\left(\boldsymbol{X}_t|\boldsymbol{Z}_{1:t}\right) = \frac{p\left(\boldsymbol{Z}_t|\boldsymbol{X}_t\right) p\left(\boldsymbol{X}_t|\boldsymbol{Z}_{1:t-1}\right)}{\int p\left(\boldsymbol{Z}_t|\boldsymbol{X}_t\right) p\left(\boldsymbol{X}_t|\boldsymbol{Z}_{1:t-1}\right) d\boldsymbol{X}_t} \quad (5)$$

Fig. 2 depicts the steps of this procedure. The posterior at time (i.e. frame) $t-1$, $p\left(\boldsymbol{X}_{t-1}|\boldsymbol{Z}_{1:t-1}\right)$ (Fig. 2(a)) is propagated to the priori at time t (Fig. 2(b)). At the very first frame no prior assumptions are made and $p\left(\boldsymbol{X}_0|\boldsymbol{Z}_0\right)$ is set to a uniform distribution. The predicted pdf is obtained applying the motion model on the priori and then marginalizing on $\boldsymbol{X}_{t-1}$ (eq. 4). Since in complex scenes correct motion model is unknown [59], we apply a zero-order function with Gaussian noise of fixed covariance, i.e. the priori is convolved with white noise which has the only effect of increasing its covariance (Fig. 2(c)). The predicted pdf is then exploited for driving the MS-PW: differently from the case of single images, where the proposal of the first stage q_0 is uniform, in the case of videos $q_0\left(\boldsymbol{X}_t\right)$ is obtained by applying a *quasi-random sampling* [60] to the predicted distribution:

$$q_0\left(\boldsymbol{X}_t\right) = \beta \cdot p\left(\boldsymbol{X}_t|\boldsymbol{Z}_{1:t-1}\right) + (1-\beta) \cdot U\left(\boldsymbol{X}_t\right) \quad (6)$$

where β adjusts the amount of random sampling. Adding a uniform distribution on top of the predicted is useful for enabling the algorithm to detect new objects entering the scene or appearing from occluding objects (the effect is depicted by the yellow particles on Fig. 2(d)). Starting from such proposal, the MS-PW method described in Section 3.2 obtains a set of successful classifications (Fig. 2(e)) and estimates the likelihood $p\left(\boldsymbol{Z}_t|\boldsymbol{X}_t\right)$ (Fig. 2(f)): differently from the case of single images, the final object detection is not obtained from the modes of the likelihood, but from the modes of the posterior pdf (Fig. 2(g),2(h) and eq. 5). The whole prediction-update process is very fast since it is computed on compact mixtures of Gaussians: indeed, the number of components is fairly limited, since it approximately corresponds to the number of target objects in the image.

4 Experimental Results

4.1 Brief description of exemplar classifiers

In section 4.4 we will compare the MS-PW and the SW detection using four different classifiers and features which are here briefly introduced: covariance-matrix-based pedestrian detector [9] (CovM), Haar-like-features based face detector [10] (HLF), Integral-channel-features based pedestrian detector [20] (FPDW), Histogram-of-oriented-gradients based pedestrian detector [8] (HOG). Further details can be found on the original papers. In any of the four cases, MS-PW and SW share exactly the same classifier, feature, parameters configuration and trained model.

4.1.1 CovM

The authors propose to adopt a rejection cascade of LogitBoost classifiers [45], exploiting linear logistic regressors as weak learners: each weak learner operates on a rectangular and axis-oriented sub-window of the window to classify. The employed cue is given by the covariance matrix of a 8-dimensional set of features F (defined over each pixel of an image I) which account for pixel coordinates and first- and second-order derivatives of the image.

The covariance matrix of the set of features F can be computed on any sub region of I and can be used as *covariance descriptor* in the classifier. In particular, since the sub-windows associated with each weak learner are rectangular and axis-oriented, the covariance descriptors can be efficiently computed using *integral images* [10], [61]. With this artifice the computation of the descriptor of any weak learner is obtained in constant time, regardless of the size of the sub-window. However, the 8x8 covariance matrix obtained belongs to the Sym_d^+ space (symmetric and positive semi-definite matrices), that is a Riemannian manifold and in order to apply any traditional classifier the matrix has to be mapped over an Euclidean space through the inverse of the exponential mapping [9].

An important advantage of the covariance descriptors is its relatively low degree of sensitivity to small translations and scale variations. The cascade of LogitBoost classifiers of [9] trained on the INRIA pedestrian dataset shows a radius of the region of support of approximately 10-15% of the window size in position and 20% in scale. To confirm this hypothesis we provide some tests (Fig. 3): we cropped 100 images from the INRIA test set, with the characteristic of having one pedestrian of 50x150 pixels centered in the cropped image and an average of 2.5 other pedestrians per image (acting as distractors), at random positions and scales. For each image, we computed the value of $R(w)$ with a pixel stride of 1 and a scale stride of 1.01 and then averaged the R over all images.

Fig. 3(f) clearly shows the plateau in the center of the window, that represents the region of support of the classifier, surrounded by its basin of attraction. The two local maxima on the side are due to the regions of support of the other pedestrians that, as expectable, tend to cluster closeby. The region of support of Fig. 3, scaled accordingly to the reference pedestrian size of 32x96 of the INRIA pedestrian dataset, shows a diameter of

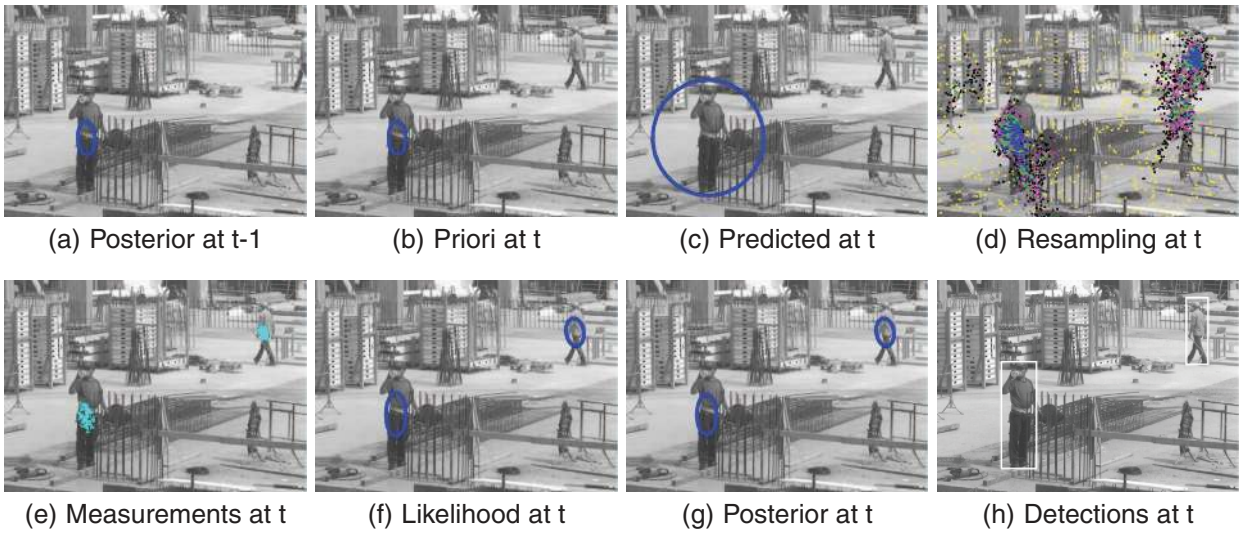


Fig. 2: MS-PW in the context of Bayesian recursive filtering. The blue ellipses in (a,b,c,f,g) represent the Gaussian components of the mixture of a pdf: each ellipse discards the scale dimension (w_s) and depicts the covariance only on the location (w_x,w_y). Alternatively, the white rectangles in (h) represent the object detections at the proper scale and position obtained from the modes of the posterior. In (d) the coloring order of the particles across the MS-PW stages is yellow, black, magenta, green and blue. The man on the upper-right corner, that enters the scene at time t, is completely outside the influence of the predicted pdf; nevertheless, the uniform component of quasi random sampling enables his detection. In (e), the cyan dots depict the particles that yielded successful detections (set S, line 18, Alg. 1).

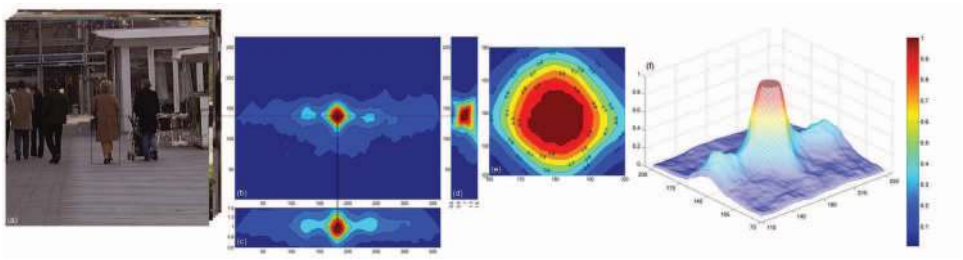


Fig. 3: Region of support and basin of attraction for the cascade of LogitBoost classifiers [9] trained on INRIA pedestrian dataset. (a) a sample taken from the set of 100 cropped images: there is a centered pedestrian of 50x150 pixels and an average of 2.5 other pedestrians at other positions and scales; (b-e) response (computed with eq. 2) of the classifier averaged on the 100 images: (b) fixed scale w_s (equal to 50x150), sliding w_x, w_y; (c) fixed w_x (equal to x of window center), sliding w_s and w_y; (d) fixed w_y (equal to y of image center), sliding w_x and w_s; the central region of (b) is enlarged in (e) and plotted in 3D in (f).

approximately 9 pixels, confirming that the pixel stride of 6 proposed by [9] is reasonable.

4.1.2 HLF

The well-known Viola-Jones face detector adopts a set of nonadaptive Haar-like features, which are reminiscent of Haar basis functions proposed in [62]. Using a base resolution on the detector of 24x24 (extensive analysis of the influence of this value on the performance has been proposed in [25]) this leads to 45,396 features, which is over-complete and a rich dictionary of simple features. The features are computed very rapidly with integral images. For each layer of the cascade, AdaBoost iteratively constructs a weighted linear combination of weak classifiers, where each weak classifier is simply made by thresholding one feature value (called "single perceptron" in [10]). Viola and Jones proposed to use a cascade made of 32 layers, with a total of 4,297 features (among the original 45,396).

The region of support of the Viola-Jones face detector with Haar-like feature is depicted in Fig. 4 and appears to be much more compact w.r.t. that reported in Fig. 3: this can be ascribed to the different employed feature and to the different modeled object: indeed, a frontal face has a more descriptive shape w.r.t. the pedestrian observed from a generic point of view. For this reason, the related papers often suggest a pixel stride as low as 1 [26], [27], [32].

4.1.3 FPDW

The idea behind pedestrian detection with integral channel features [21] is to put together the descriptiveness of multiple registered image channels (gradient histograms, gradient magnitude and LUV color channels), and the computational efficiency provided by integral images. The authors in [21] propose the use of a soft cascade, where a threshold is used after evaluation of every weak classifier. Depending on the dataset used for testing, this

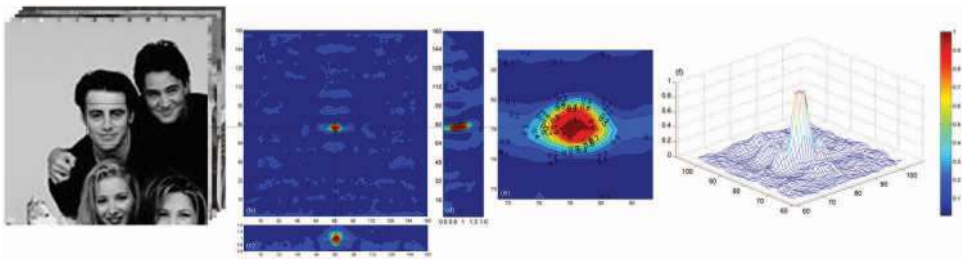


Fig. 4: Region of support and basin of attraction for the cascade of AdaBoost classifiers [7]. (a) a sample taken from the set of 100 cropped images from the MIT+CMU dataset: there is a centered face of 38x38 pixels and an average of 1.8 other faces at other positions and scales; (b-e) response (computed with eq. 2) of the classifier averaged on the 100 images: (b) fixed scale w_s (equal to 38x38), sliding w_x, w_y; (c) fixed w_x (equal to x of window center), sliding w_s and w_y; (d) fixed w_y (equal to y of window center), sliding w_x and w_s; the central region of (b) is enlarged in (e) and plotted in 3D in (f).

algorithm represents the state-of-the art or the second best algorithm (second to [44]) in pedestrian detection. In [20], the authors propose an approach to speed up the algorithm by approximately a factor 10, reducing the computation on the construction of the feature pyramid, suffering only a very minor loss in detection accuracy. Such optimization is totally complementary to our MS-PW method, that optimizes the search in the state space, and both can be combined together yielding a multiplication of the speed up factors w.r.t. the [21] detector, as it will be demonstrated in section 4.4. Fig. 5 shows the region of support for such algorithm, computed according to the R for soft cascades described in section 3.1.

4.1.4 HOG

In order to demonstrate that the multi-stage method, if fed through appropriate confidence measures, is not limited to ensemble classifiers but can be applied to on any real-valued confidence classifier, we also tested the popular pedestrian classifier proposed in [8] that apply a SVM classifier to Histogram of Oriented Gradients (HOG): this feature counts occurrences of gradient orientation in localized portions of an image and happens to be particularly suited for the task of pedestrian detection. The computation of the histograms is performed on a dense grid of uniformly spaced cells and overlapping local contrast normalization is used for improved accuracy; the SVM classifier can either be linear or use other kernels (e.g., Radial Basis Function). Since several works still recently take inspiration and move forward from the detection architecture proposed by [8] ([44] is among the most successful examples), we believe that demonstrating the feasibility of the Multi-Stage detection onto HOG-SVM classifiers is of particular significance. Fig. 6 shows the region of support and corresponding basin of attraction for this classifier computed according to eq. 3.

4.2 Benchmark and evaluation metrics

The experiments we propose have a twofold bottom line, that we consider as success indicators: demonstrating that when MS-PW is configured to operate at the same detection accuracy of SW, it exhibits lower detection time; conversely, when it is configured to operate with the same detection time of SW, it yields a higher detection accuracy. These different operating points are achieved configuring SW to work at its best (using the parameters provided in the original works) and then tuning the number of particle windows of MS-PW to reach the desired operating point (either same accuracy or same speed of SW). Within this evaluation process, the classifiers are left totally untouched between the two detection approaches. To reach both advantages together (faster AND better accuracy) a different operating point should be selected just employing a number of particle windows in between the two extreme cases: such working point is a trade-off between the two and provide faster computation and better accuracy but both of them at a lower degree with respect to the extremes.

The experimental results are obtained on the benchmark reported in Table 2: the whole list of datasets is public and comprises ground truth bounding-box annotations. In the case of single images, we have selected four different datasets, three for pedestrians and one for faces. Fig. 7 shows a few samples (and the detection produced with CovM and HLF methods for pedestrian and face, respectively) from each of the four datasets.

Regarding pedestrian detection, one of the most widely used dataset is INRIA [8], since it contains very different pedestrian sizes, even within the same frame, has some low quality images, and has very different frame sizes. On this dataset we have performed comparative tests between SW and MS-PW on all the three classifiers (CovM, FPDW, HOG) working on the two above-mentioned operating points. CovM has been also tested on two other image datasets, namely Graz02 [63] and our CWSi (Construction Working Sites image)[3]. In Graz02 the SW-based detection works rather well even with a very coarse striding in location and scale, i.e. a very low number of windows: this can be ascribed to the visual properties of the dataset, that seem to be favorable to pedestrian detection (e.g. good image

2. http://www.emt.tugraz.at/~pinz/data/GRAZ_02/person.zip

3. CWSi and CWSv datasets and annotation can be publicly downloaded at http://imagelab.ing.unimore.it/visor

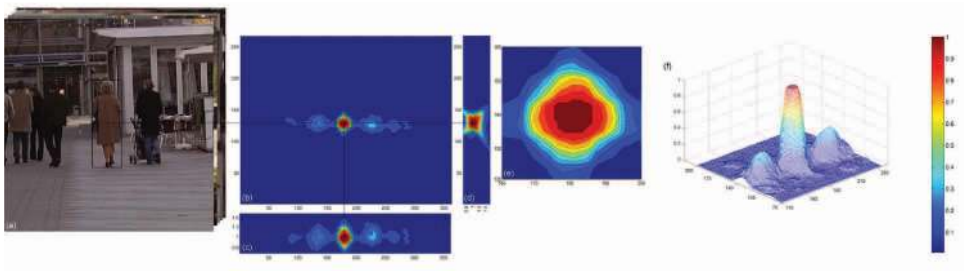


Fig. 5: Region of support and basin of attraction for the FPDW approach [20]. Same training set, data and description of Fig. 3.

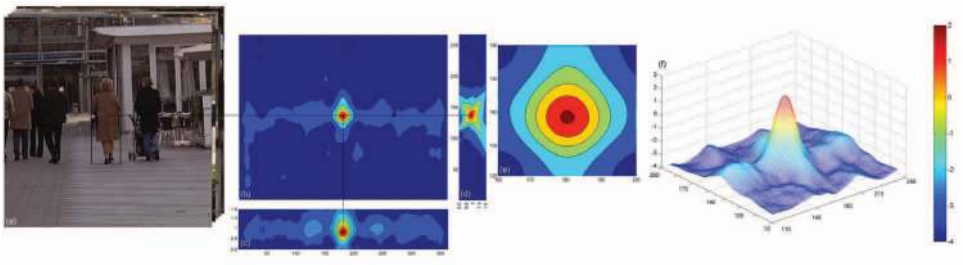


Fig. 6: Region of support and basin of attraction for the HOG+SVM approach [8]. Same training set, data and description of Fig. 3. In this case, the graphs report the margin of SVM.

		# images	Image size	# objects	Object size	Avg object/im.
Tests on Images for Pedestrian Detection	INRIA [8]	311	333x531-1280x960	582	32x80-320x800	1.87
	Graz02[2] [63]	311	640x480	777	17x42-255x639	2.50
	CWSi[3]	300	800x600	781	22x55-181x386	2.60
Tests on Images for Face Detection	MIT+CMU [10]	130	60x75-1280x1024	511	14x14-486x486	3.93
Tests on CWSv Videos	Video 1	300	800x600 @ 1 fps	413	22x55-181x386	1.38
	Video 2	300		583		1.94
	Video 3	300		966		3.22
	Video 4	300		975		3.25
	Video 5	300		1180		3.93
	Video 6	300		641		2.14
	Video 7	300		969		3.23

TABLE 2: Benchmark.

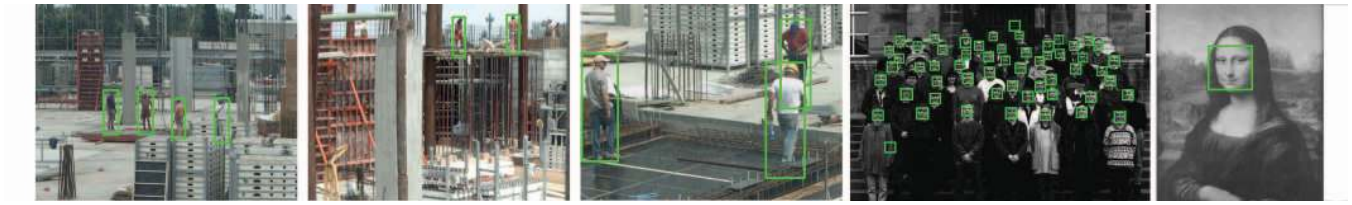
quality, constant frame size, fairly detached pedestrians, etc.). For this reason the advantage of our MS-PW will not be as evident as in the two other datasets, that are definitely more challenging. CWSi contains similar challenging conditions than INRIA but also includes challenging backgrounds, distractors and people squatted or occluded by pillars or scaffoldings (see Fig. 7(b)). For the evaluation of face detection, we have employed the same dataset used by HLF [10] and many other successive works (see Tab. 1 in Section 2). In total, we evaluated more than a thousand of images with more than 2500 objects at different size.

Regarding the videos, we performed experiments on CWSv (Construction Working Sites video) dataset[3], made of 7 videos for a total of 2100 frames, recorded in very different conditions of construction sites scenarios; similarly to CWSi, even this video dataset is seriously challenging, with the addition of compression artifacts and an average of 13.1 entrances/exits of pedestrians per video and some people grouping (up to 8 close-by people within a frame). All the tests on videos have been carried out using CovM classifier as a base and, considering only the cue of appearance exploited by the classifiers, with no use of additional cues, such as motion, geometry, depth or other priors.

The accuracy of object detection is measured at object level in terms of the PASCAL threshold defined in the PASCAL VOC challenge [64] which evaluates the ratio between the intersection and the merge of the bounding box found by the detector with the bounding box in the ground truth. This value is thresholded at T, that is set to 0.5 in most of our tests as well as in typical object detection evaluations; we also perform detection tests at varying values of T, in order to better evaluate the localization accuracy of the detection of MS-PW w.r.t. SW. Throughout all tests, multiple detections of the same ground-truthed person, as well as a single detection matching multiple ground-truthed people, are affecting the performance in terms of false positives and false negatives respectively.

In this analysis we employ *Detection Error Trade-off (DET)* curves [65], that are typically preferred to the ROC

(a) Results of MS-PW with CovM on INRIA (1st, 2nd, 3rd image) and Graz02 (4th, 5th image) dataset.



(b) Results of MS-PW with CovM on CWSi dataset (1st, 2nd, 3rd image) and with HLF on CMU dataset (4th, 5th image).

Fig. 7: Some examples of the different image datasets and MS-PW detection results.

curves for evaluating object detection results [8], [16]. A DET curve reveals how Miss Rate (MR, i.e. the reciprocal of the detection rate) changes with the rate of false positives, varying a system parameter, e.g. the detection confidence. It is possible to measure either *False Positives Per Window* (FPPW) or *Per Image* (FPPI); as stated in [34], the former is more appropriate for evaluating the object classifiers, while the latter better evaluates the detection as a whole, which is more appropriate in our case. Accuracy can also be roughly evaluated through single-valued scalar quantities, representing cumulative measurements: typical choices are the Area Under the ROC Curve (ROC-AUC) and the Average Precision (AP) [64], that better emphasizes the detection accuracy w.r.t. the detection confidence. We employed the AP, since the PASCAL VOC challenge has preferred it to the ROC-AUC, starting from the 2007 edition [66]. Additionally, we also report the miss rate at the reference value FPPI=1.

4.3 Parameters' analysis

We have tested MS-PW under two different configurations: a comprehensive and flexible configuration, employing all the parameters described in Alg. 1, and a simplified configuration, reducing the number of parameters just to a minimal and simplified set. The comprehensive configuration was thoroughly tested on CovM and HLF classifiers; the reduced configuration on FPDW and HOG. The number of windows for the SW-approach depends on pixel stride, scale stride, and frame size, for the datasets that do not have constant frame size (as in the case of INRIA and MIT+CMU). Conversely, the number of particle windows in MS-PW does not depend on any of those parameters, but only on m, the number of stages employed for detection, and on the law that regulates the decrease of the number of pw from stage to stage, as described in Section 3.2. The decrease follows an exponential law: the comprehensive configuration uses $N_i = NP \cdot e^{\gamma \cdot (i-1)}$, with $i = 1 \dots m$, where NP represents the initial number of particle windows (i.e., N_1) which remains fixed independently on the image size (we configured $m = 5$ and $\gamma = 0.44$). The minimal configuration instead follows $N_i = \frac{NP}{2^i}$ and m is bounded by the stage that reaches zero particle windows.

For the comprehensive configuration, a similar exponential trend is also applied respectively to λ_i, the exponent of the classifier response R, and to Σ_i, the covariance of the Gaussian kernels. The values at the first stage are $\lambda_1 = 0.1$ and $\Sigma_1 = diag(8, 16, 0.26)$ for CovM and $\lambda_1 = 0.1$ and $\Sigma_1 = diag(5, 5, 0.26)$ for HLF; the γ value are respectively 1 and -0.66. In any case, the initial value of Σ depends on the size of the region of support of the classifier. Instead, the reduced configuration simplifies the parameters by setting $\lambda_i = 1$, $\forall i$, and Σ_i constant for all i and equal to the size of the region of support. In both configurations, the adaptation rate α is set to 1, so that the update of the proposal at stage i depends completely on the measurement function $p_{i-1}(Z|X)$.

4.4 Evaluation of Accuracy and Speed on Images

For each tested classifier, on the SW detection we employ pixel and scale stride values suggested in the original papers, that are usually aimed at obtaining the best trade-off between speed and accuracy. In addition, for the HOG classifier, the only non-cascaded classifier, we compute the SW at an additional operating point (with higher scale stride, see rows 11-13 of Table 3) that is often used in video surveillance scenarios when necessary to reduce computation as much as possible.

The results are reported in Fig. 8 and Table 3. The PASCAL threshold is set to $T = 0.50$. For each SW detection measurement, MS-PW is tuned to work at two different operating points, i.e. to obtain approximately the same

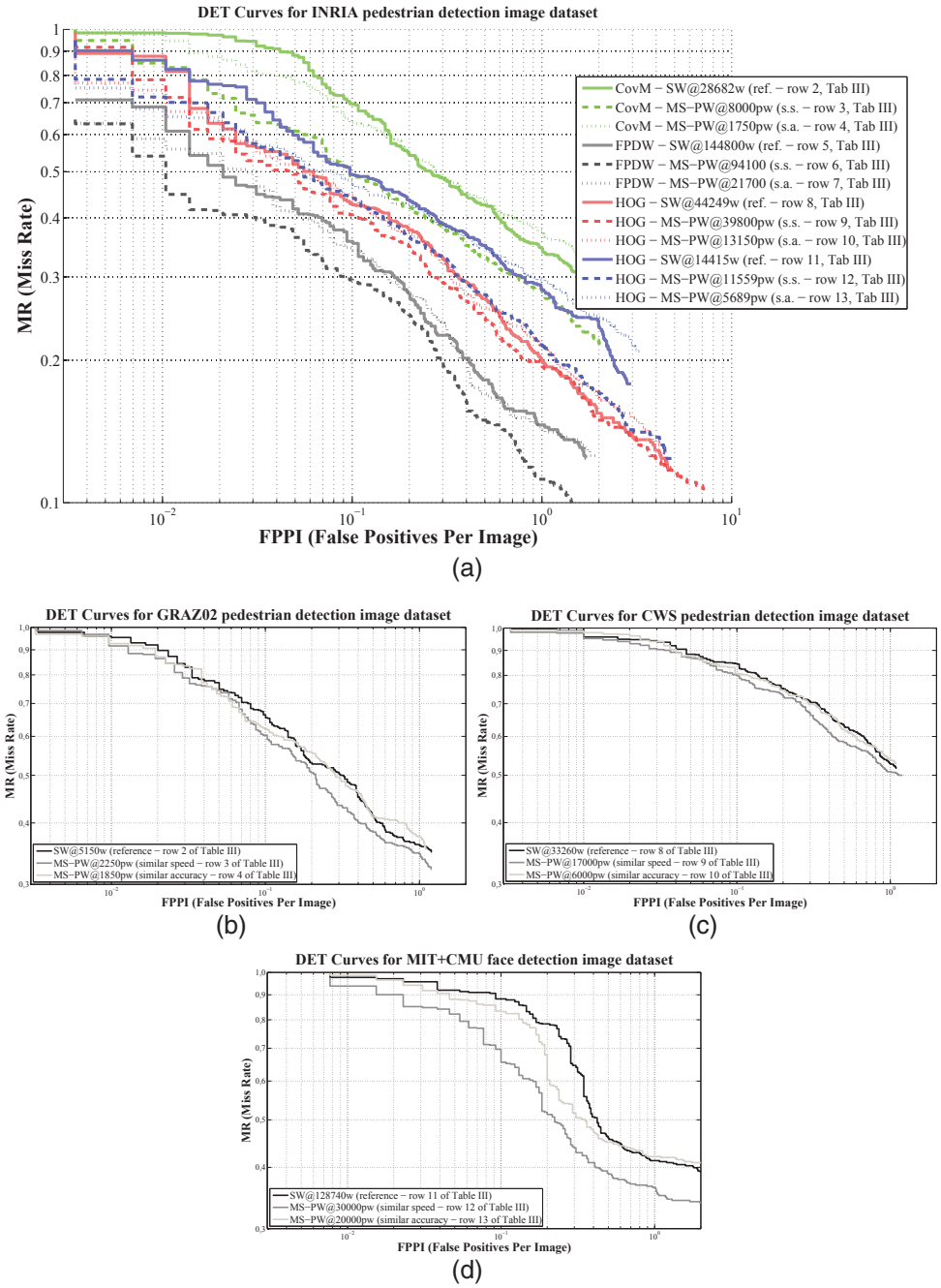


Fig. 8: DET curves comparing SW and MS-PW approaches on different image datasets. The labels also report the averaged number of windows (w) or particle windows (pw) employed by the detectors. For MS-PW curves, the lighter one is configured to yield similar Average Precision as the SW curve, while the darker runs at approximately same speed of SW.

speed or the same accuracy (in terms of AP) of SW. Fig. 8 shows that MS-PW with similar speed is generally more accurate (i.e., the DET curve is lower) than SW, for every classifier. This advantage is emphasized on the complex datasets. Indeed, as reported in Table 3 - sixth column, the AP gain obtained by using MS-PW with CovM classifier goes from 8.8% in the case of GRAZ02, to 21.9% in the case of INRIA. Still on INRIA dataset, MS-PW gains 4.73% in AP w.r.t SW with the FPDW algorithm, and respectively 2.33% and 10.01% with the HOG algorithm. In this latter case, the tests highglight as the MS-PW accuracy has better scalability to lower number of particles rather than SW. When considering MS-PW at similar AP, the speedup in time (last column of Table 3) reaches 3.81 for INRIA with CovM approach, which means almost a fourth of the time required. In the case of FPDW algorithm the speedup is reduced to 2.02%, while in the two operating points of HOG we reached 2.01 and 1.86, respectively. We prefer to report time speedup instead of time absolute values since these latter measurments would be strongly implementation-dependent, i.e. they would be determined by the em-

ployed PC, the software toolchain and the quality of the implemented code.

One key contribution of our multi-stage approach is to progressively refine the sampling of particle windows to obtain accurate localization of the detections. To demonstrate this we have further analyzed the behavior of MS-PW with respect to SW when the PASCAL threshold T gets changed. Fig. 9(a) shows the comparison of MS-PW with SW at $T = 0.30, 0.55, 0.65$ with CovM approach. Increasing the T implies a request for higher degree of overlap between detection and ground truth bounding boxes: as expectable, together with the increase of T, the DET curves raise and the APs decrease. The remarkable point here is that the decreasing rate of AP is definitely more prominent for the SW approach rather than for MS-PW, as shown by the increasing AP gain of Fig. 9(b); in other words, the detection localization of MS-PW is higher, since it suffers in a lower degree the request for higher overlaps determined by higher Ts; this is to be ascribed to the information gain obtained through the multi-stage sampling strategy (see Fig. 10).

4.5 Evaluation of Accuracy and Speed on Videos

The experiments on videos are aimed at validating the usefulness of plugging the multi-stage particle window paradigm inside the Bayesian-recursive approach, in order to exploit the temporal coherency of the target objects. We focused on pedestrian detection with CovM approach only and made use of the CWSv test set (details in Table 2). We compared SW detection, MS-PW detection configured to yield similar accuracy and eventually MS-PW detection configured exactly in the same manner, with the further addition of being plugged into a Bayesian-recursive framework (Section 3.3). At the cost of a minor decrease in speed-up (4.7% slower on average), this latter configuration shows a strong boost in accuracy (details in Table 4 and Figure 11 that shows an exemplar DET curve measured on a video). The slight decrease of speed-up is due to the fact that, thanks to the non uniform proposal distribution (see Equation 6), a portion of the particle windows are sampled around true positives already at the very first stage: together with improving detection rates, it determines also a slight increase of the average R, and therefore of the computational burden.

	SW	**MS-PW non rec.**		**MS-PW rec.**	
	AP	AP	**Speedup**	AP (gain)	**Speedup**
V1	#w=28556	#pw=5000		#pw=5000	
	0,373	0,372	1,83	0,389 (4,18%)	1,96
V2	#w=28556	#pw=5000		#pw=5000	
	0,400	0,399	1,53	0,404 (0,97%)	1,58
V3	#w=28556	#pw=5000		#pw=5000	
	0,511	0,517	1,78	0,549 (7,60%)	1,57
V4	#w=28556	#pw=4000		#pw=4000	
	0,607	0,607	1,50	0,683 (12,49%)	1,19
V5	#w=28556	#pw=5400		#pw=5400	
	0,662	0,662	1,62	0,678 (2,36%)	1,46
V6	#w=28556	#pw=4000		#pw=4000	
	0,569	0,574	2,46	0,597 (4,85%)	2,42
V7	#w=28556	#pw=4000		#pw=4000	
	0,408	0,406	2,61	0,568 (39,39%)	2,51
Avg	0,504	0,505	**1,90**	0,553 (**9,58%**)	**1,81**

TABLE 4: Results on video datasets.

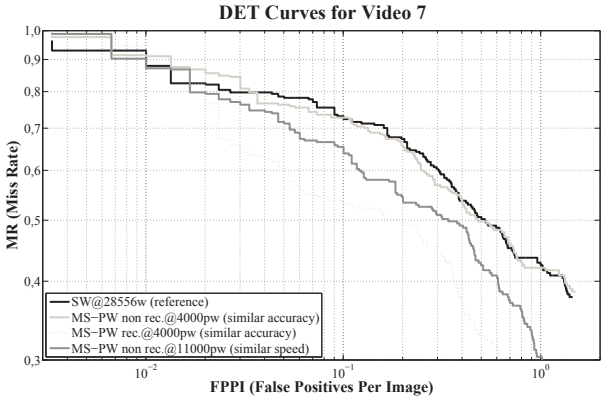


Fig. 11: DET curves for Video 7.

5 Conclusions

The work introduces a novel method for hypothesis search in problems of object detection, avoiding the drawbacks of sliding window paradigm. Through the definition of the response R of a given classifier, the proposed method exploits the presence of a basin of attraction around true positives to drive an efficient exploration of the state space, using a multi-stage sampling based strategy. The derived measurement function can be plugged in a kernel-based Bayesian filtering to exploit temporal coherence of pedestrians in videos. The use of a measurement model that is the same for all target objects, together with a quasi-random sampling allows the method to deal with objects entrances and exits. Experimental evaluation is performed on pedestrian and face detection and using very diverse types of cascaded and non-cascaded classifiers; results show that the proposed method obtains similar accuracy at lower computational load w.r.t. sliding window; conversely, working at the same computational load, the detection rate and localization are higher. The advantage of the proposed method is emphasized in complex datasets, where the sliding window approach obtains good detection results at the cost of a very large number of windows; our method can take the place of SW improving performance either in speed or in accuracy or in both of them. The proposed method yields also an increased accuracy in the localization of detections. In future works we plan to adapt MS-PW even for the training phase, specifically for efficient bootstrapping. Additionally we plan to investigate the possibility to use MS-PW on a specific feature with a specific classifier just for efficient state space exploration, independently on the eventual object detection classifier. Finally, we want to test MS-PW on part-based approaches such as [44], since every part classifier is characterized by its own basin of attraction.

	Dataset	Classifier	Approach	# windows	AP (gain)	$MR@FPPI = 1$	Speedup
2	INRIA	CovM [9]	SW	28682 (avg)/94445 (max)	0.562 (n/a)	0.340	n/a
3			MS-PW s.s.	8000 (4152/2062/1024/508/252)	0.685 (**21.9%**)	0.280	n/a
4			MS-PW s.a.	1750 (908/451/224/111/55)	0.562 (n/a)	0.380	**3.81**
5		FPDW [20]	SW	144800 (avg)/ 488500 (max)	0.825 (n/a)	0.153	n/a
6			MS-PW s.s.	94100 (47051/23530/11767/5884/...)	0.864 (**4.73%**)	0.112	n/a
7			MS-PW s.a.	21700 (10858/5430/2715/1358/...)	0.826	0.147	**2.02**
8		HOG [8]	SW	44249 (avg)/146841 (max)	0.771 (n/a)	0.203	n/a
9			MS-PW s.s.	39800 (19898/9950/4976/2488/1244/...)	0.789 (**2.33%**)	0.201	n/a
10			MS-PW s.a.	13150 (6574/3288/1644/822/411/...)	0.771 (n/a)	0.203	**2.91**
11		HOG [8]	SW	14415 (avg)/47473 (max)	0.699 (n/a)	0.283	n/a
12			MS-PW s.s.	11559 (5779/2890/1445/723/361/...)	0.769 (**10.01%**)	0.221	n/a
13			MS-PW s.a.	5689 (2844/1422/711/356/178/...)	0.699 (n/a)	0.294	**1.86**
14	Graz02	CovM [9]	SW	5150 (fixed value)	0.533 (n/a)	0.362	n/a
15			MS-PW s.s.	2250 (1696/418/103/25/6)	0.580 (**8.8%**)	0.343	n/a
16			MS-PW s.a.	1850 (1395/344/84/20/5)	0.533 (n/a)	0.345	**1.16**
17	CWSi	CovM [9]	SW	33260 (fixed value)	0.335 (n/a)	0.528	n/a
18			MS-PW s.s.	17000 (8824/4382/2176/1080/536)	0.371 (**10.9%**)	0.508	n/a
19			MS-PW s.a.	6000 (3114/1546/768/381/189)	0.335 (n/a)	0.538	**2.55**
20	MIT+CMU	HLF [10]	SW	128740 (avg)/819491 (max)	0.506 (n/a)	0.413	n/a
21			MS-PW s.s.	30000 (15572/7733/3840/1906/946)	0.606 (**19.7%**)	0.364	n/a
22			MS-PW s.a.	20000 (10381/5155/2560/1271/631)	0.506 (n/a)	0.419	**1.34**

TABLE 3: Results on image datasets for pedestrian and face detection. s.s.=similar speed, s.a.=similar accuracy. n/a = not appl.

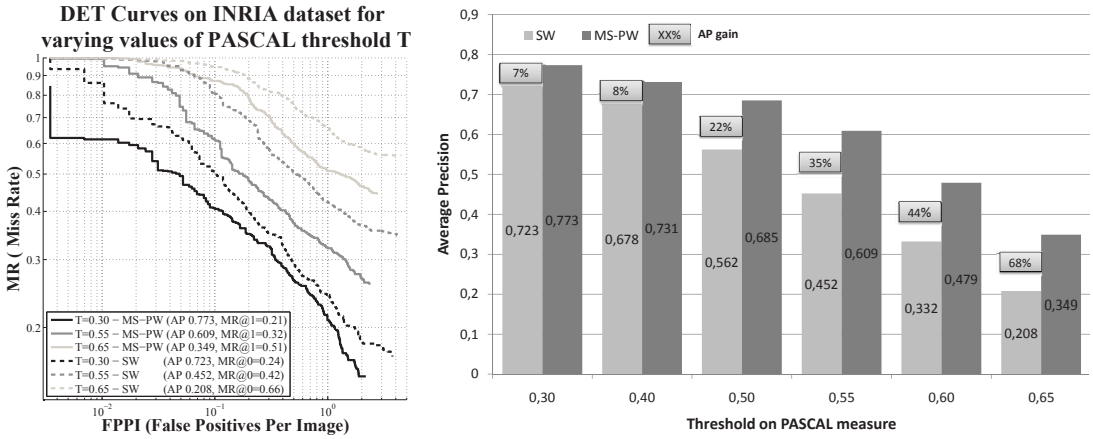


Fig. 9: Results on INRIA dataset at different values of PASCAL threshold T.

ACKNOWLEDGMENTS

This work is within the project THIS (JLS/2009/CIPS/AG/C1-028), with the support of the Prevention, Preparedness and Consequence Management of Terrorism and other Security-related Risks Programme European Commission - Directorate-General Justice, Freedom and Security. This project is also partially funded by Regione Emilia-Romagna (PRRIITT funding scheme) and Bridge129 SpA. The authors want to deeply thank Piotr Dollar from Caltech for his help in experimenting with FPDW classifier.

REFERENCES

[1] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.

[2] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *In Proc. 13th Int. Conference on Machine Learning*, 1996, pp. 148–156.

[3] E. Bauer and R. Kohavi, "An empirical comparison of voting classification algorithms: Bagging, boosting, and variants," *Machine Learning*, vol. 36, pp. 105–139, 1999.

[4] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, 20-25 2005, pp. 524 – 531 vol. 2.

[5] M. Enzweiler and D. Gavrila, "Monocular pedestrian detection: Survey and experiments," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 12, pp. 2179 –2195, dec. 2009.

[6] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97 – 136, 1980.

[7] P. Viola, M. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *IJCV*, vol. 63, no. 2, pp. 153–161, 2005.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, vol. 1, June 2005, pp. 886–893 vol. 1.

[9] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on riemannian manifolds," *IEEE T-PAMI*, vol. 30, no. 10, pp. 1713–1727, Oct. 2008.

[10] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, pp. 137–154, 2004.

[11] R. Verschae, J. Ruiz-del Solar, and M. Correa, "A unified learning framework for object detection and classification using nested cascades of boosted classifiers," *Machine Vision and Applications*, vol. 19, pp. 85–103, 2008.

[12] S. Brubaker, J. Wu, J. Sun, M. Mullin, and J. Rehg, "On the design of cascades of boosted ensembles for face detection," *International Journal of Computer Vision*, vol. 77, pp. 65–86, 2008.

(a) (b)

Fig. 10: Detection with SW (a) and with MS-PW (b) on an example image taken from INRIA test dataset. MS-PW shows a remarkably higher localization of detections. Both runs generated 1 false positive in the image, that has been removed for clarity of depiction.

[13] L. Itti and C. Koch, "A saliency-based search mechanism for overt and covert shifts of visual attention," *Vision Research*, vol. 40, no. 10-12, pp. 1489 – 1506, 2000.

[14] L. Zhang, M. H. Tong, and G. W. Cottrell, "Information attracts attention: A probabilistic account of the cross-race advantage in visual search," in *in Proceedings of the 29th Annual Cognitive Science Conference*, 2007.

[15] N. Butko and J. Movellan, "Optimal scanning for faster object detection," in *CVPR 2009. IEEE Conference on*, 2009, pp. 2751 – 2758.

[16] W. Zhang, G. Zelinsky, and D. Samaras, "Real-time accurate object detection using multiple resolutions," in *ICCV 2007. IEEE Conference on*, 2007, pp. 1 –8.

[17] A. Oliva and A. Torralba, "Chapter 2 building the gist of a scene: the role of global image features in recognition," in *Visual Perception - Fundamentals of Awareness: Multi-Sensory Integration and High-Order Perception*, ser. Progress in Brain Research, S. Martinez-Conde, S. Macknik, L. Martinez, J.-M. Alonso, and P. Tse, Eds. Elsevier, 2006, vol. 155, Part 2, pp. 23 – 36.

[18] M. Pedersoli, J. Gonzàlez, A. D. Bagdanov, and J. J. Villanueva, "Recursive coarse-to-fine localization for fast object detection," in *Computer Vision – ECCV 2010*, ser. Lecture Notes in Computer Science, vol. 6316, 2010, pp. 280–293.

[19] F. Fleuret and D. Geman, "Coarse-to-fine face detection," *International Journal of Computer Vision*, vol. 41, no. 1/2, pp. 85–107, 2001.

[20] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *British Machine Vision Conference (BMVC)*, 2010, pp. 1–11.

[21] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *British Machine Vision Conference (BMVC)*, 2009.

[22] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE T-PAMI*, vol. 25, no. 5, pp. 564–575, 2003.

[23] B. Han, Y. Zhu, D. Comaniciu, and L. S. Davis, "Visual tracking by continuous density propagation in sequential bayesian filtering framework," *IEEE T-PAMI*, vol. 31, no. 5, 2009.

[24] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," *IJCV*, vol. 29, no. 1, pp. 5–28, 1998.

[25] R. Lienhart, A. Kuranov, and V. Pisarevsky, "Empirical analysis of detection cascades of boosted classifiers for rapid object detection," in *Pattern Recognition*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2003, vol. 2781, pp. 297–304.

[26] B. Froba and A. Ernst, "Face detection with the modified census transform," in *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, 17-19 2004, pp. 91 – 96.

[27] J. Wu, M. D. Mullin, and J. M. Rehg, "Linear asymmetric classifier for cascade detectors," in *ICML*, 2005, pp. 988–995.

[28] T. B. Dinh, V. B. Dang, D. A. Duong, T. T. Nguyen, and D.-D. Le, "Hand gesture classification using boosted cascade of classifiers," in *Research, Innovation and Vision for the Future, 2006 International Conference on*, feb. 2006, pp. 139 – 144.

[29] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2, 2006, pp. 1491 – 1498.

[30] S. Paisitkriangkrai, C. Shen, and J. Zhang, "Fast pedestrian detection using a cascade of boosted covariance features," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 8, pp. 1140 –1151, aug. 2008.

[31] S. Munder and D. Gavrila, "An experimental study on pedestrian classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1863–1868, 2006.

[32] W. Zhang, R. Tong, and J. Dong, "Boosted cascade of scattered rectangle features for object detection," *Science in China Series F: Information Sciences*, vol. 52, pp. 236–243, 2009.

[33] R. N. Hota, K. Jonna, and P. R. Krishna, "On-road vehicle detection by cascaded classifiers," in *COMPUTE '10: Proceedings of the Third Annual ACM Bangalore Conference*. New York, NY, USA: ACM, 2010, pp. 1–5.

[34] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *CVPR*, June 2009, pp. 304–311.

[35] L. Breiman, J. Friedman, C. Stone, and R. Olshen, *Classification and Regression Trees*. CRC Press, 1984.

[36] T. Ojala, M. Pietikinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51 – 59, 1996.

[37] R. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, pp. 179–188, 1936.

[38] C. Wohler and J. Anlauf, "An adaptable time-delay neural-network algorithm for image sequence analysis," *Neural Networks, IEEE Transactions on*, vol. 10, no. 6, pp. 1531 –1536, nov 1999.

[39] A. Lehmann, B. Leibe, , and L. V. Gool, "Feature-centric efficient subwindow search," in *ICCV*, October 2009.

[40] E.-J. Ong and R. Bowden, "A boosted classifier tree for hand shape detection," *Automatic Face and Gesture Recognition, IEEE International Conference on*, vol. 0, p. 889, 2004.

[41] Z. Zhang1, M. Li, S. Z. Li, and H. Zhang, "Multi-view face detection with floatboost," *Applications of Computer Vision, IEEE Workshop on*, vol. 0, p. 184, 2002.

[42] P. Viola and M. Jones, "Fast and robust classification using asymmetric adaboost and a detector cascade," in *Neural Information Processing Systems (NIPS)*, vol. 14, 2001.

[43] M. Arenas, J. Ruiz-del Solar, and R. Verschae, "Detection of aibo and humanoid robots using cascades of boosted classifiers," in *RoboCup 2007: Robot Soccer World Cup XI*, ser. Lecture Notes in Computer Science, 2008, vol. 5001, pp. 449–456.

[44] P. F. Felzenszwalb, R. B. Girshick, and D. Mcallester, "Cascade object detection with deformable part models," in *Proc. of IEEE Int'l Conference on Computer Vision and Pattern Recognition*, 2010.

[45] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *Annals of Statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[46] S. Brubaker, M. Mullin, and J. Rehg, "Towards optimal training of cascaded detectors," in *Computer Vision ECCV 2006*, ser. Lecture Notes in Computer Science, A. Leonardis, H. Bischof, and A. Pinz, Eds. Springer Berlin / Heidelberg, 2006, vol. 3951, pp. 325–337.

[47] C. Wojek and B. Schiele, "A performance evaluation of single and multi-feature people detection," in *DAGM Symp. on Patt Rec*, 2008, pp. 82–91.

[48] P. Sabzmeydani and G. Mori, "Detecting pedestrians by learning shapelet features," *CVPR*, pp. 1–8, 2007.

[49] D. M. Gavrila and S. Munder, "Multi-cue pedestrian detection and

tracking from a moving vehicle," *Int. J. Comput. Vision*, vol. 73, no. 1, pp. 41–59, 2007.
[50] J. Tao and J.-M. Odobez, "Fast human detection from videos using covariance features," in *Workshop on VS at ECCV*, 2008.
[51] A. Ess, B. Leibe, K. Schindler, and L. van Gool, "Robust multi-person tracking from a mobile platform," *IEEE T-PAMI*, vol. 31, no. 10, pp. 1831–1846, 2009.
[52] D. Hoiem, A. A. Efros, and M. Hebert, "Putting objects in perspective," *IJCV*, vol. 80, no. 1, pp. 3–15, 2008.
[53] G. Gualdi, A. Prati, and R. Cucchiara, "Covariance descriptors on moving regions for human detection in very complex outdoor scenes," in *ACM/IEEE ICDSC*, Aug. 2009.
[54] C. Wojek, G. Dorkó, A. Schulz, and B. Schiele, "Sliding-windows for rapid object class localization: A parallel technique," in *DAGM Symp. on Patt Rec*, 2008.
[55] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Efficient subwindow search: A branch and bound framework for object localization," *IEEE T-PAMI*, vol. 31, 2009.
[56] B. Han, D. Comaniciu, Y. Zhu, and L. S. Davis, "Sequential kernel density approximation and its application to real-time visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 1186–1197, 2008.
[57] O. Lanz, "An information theoretic rule for sample size adaptation in particle filtering," sep. 2007, pp. 317 –322.
[58] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *ECCV (2)*, 2006, pp. 428–441.
[59] B. Han, D. Comaniciu, Y. Zhu, and L. Davis, "Incremental density approximation and kernel-based bayesian filtering for object tracking," in *CVPR*, 2004.
[60] V. Philomin, R. Duraiswami, and L. Davis, "Quasi-random sampling for condensation," in *ECCV*, '00, pp. 134–49.
[61] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *In Proc. 9th ECCV*, 2006, pp. 589–600.
[62] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE Trans. PAMI*, vol. 23, no. 4, pp. 349–361, 2001.
[63] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer, "Generic object recognition with boosting," *IEEE T-PAMI*, vol. 28, no. 3, pp. 416–431, 2006.
[64] J. Ponce, T. Berg, M. Everingham, D. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, C. Russell, A. Torralba, C. Williams, J. Zhang, and A. Zisserman, *Dataset issues in object recognition*. Springer, 2006, p. 2948.
[65] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The det curve in assessment of detection task performance," 1997, pp. 1895–1898.
[66] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

Giovanni Gualdi (MS '03, PhD '10) is currently freelance consultant in computer vision and collaborates with ImageLab, University of Modena and Reggio Emilia (Italy). He is main inventor of 2 U.S. patents and author of more than 20 papers in international journals and conference proceedings. In 2002-2004 Giovanni served as research associate in the CVRR Lab at University of California, San Diego and in the MMSL in Hewlett Packard Labs (Palo Alto, California), addressing vision-based object tracking in sensor networks. In 2006-2010, during his PhD, Giovanni's main focus was on video surveillance in mobile scenarios, with a particular attention to object detection.

Andrea Prati (MS '98, Phd '01) is an Associate Professor currently at the Faculty of Regional and Urban Planning of University IUAV of Venice. He collaborates to research projects at regional, national and European level. Andrea Prati is author or co-author of more than 100 papers in national and international journals and conference proceedings, he has been invited speaker, organizer of workshops and journal's special issues, reviewers for many international journals in the field of computer vision and multimedia. He has also been the program chair of the 14th International Conference on Image Analysis and Processing (ICIAP), held in September 2007 in Modena and of ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC) 2011 and 2012. HE is a senior member of IEEE, and member of ACM and GIRPR (IAPR Italy).

Rita Cucchiara (MS '89, PhD '92) is Full Professor at Univ. of Modena and Reggio Emilia since 2005. Since 2008 is deputy Dean of the Faculty of Engineering and heads the Imagelab laboratory (http://imagelab.ing.unimore.it). Since 2010 she is Scientific Responsible of the ICT Platform in the High Techn. Network of Emilia Romagna. Her research interests regard computer vision, pattern recognition and multimedia systems. She is responsible of many research projects (national, and EU projects) especially in people surveillance, urban security, human-centered multimedia search of images and videos. She is author of more than 200 papers on journals and international proceedings, and she acts as reviewer for several international journals. She is in the EB of MTAP and MVA Journals, chair of several workshops on surveillance, Track chair in ICPR2012, general chair of ICIAP 2007 and AI*IA 2009. She is a member of IEEE Computer Society and ACM and IAPR. Since 2006 she is a fellow of IAPR.