Small Target Detection combining Foreground and Background Manifolds

Sébastien Razakarivony
SAGEM-SAFRAN Group, University of Caen
72-74, rueTour Billy, 95101 Argenteuil, France
sebastien.razakarivony@sagem.com

Frédéric Jurie
University of Caen
B.d. Maréchal Juin, 14000 Caen, France
frederic.jurie@unicaen.fr

Abstract

*This paper focuses on the detection of small objects (*e.g. *vehicules in aerial images) on complex backgrounds (*e.g. *natural backgrounds). A key contribution of the paper is to show that, in such situations, learning a target model and a background model separately is better than training a unique discriminative model. This contrasts with standard object detection approaches for which objects vs. background classifiers use the same types of visual features/models for both. The second contribution lies in the use of manifold learning approaches to build these models. The proposed detection algorithm is validated on the publicly available OIRDS dataset, on which we obtain state-of-the-art results.*

1 Introduction

Contrasting with most of the recent papers on object detection – which address the detection of daily life objects in high quality images [5, 6] – this paper focuses on the detection of small rigid targets (such as vehicles), in any arbitrary position, on complex textured backgrounds (see Fig. 1) . The task is made even more difficult by the fact that some objects can be camouflaged, and because it is often difficult to have large training sets as getting images of the desired targets in real condition is usually costly. Finally, object's context in image (*i.e.* the pixels surrounding the object) is not strongly correlated with the object itself.

State-of-the-art methods for object detection rely – in general – on the use of a discriminative classifier trained to learn class boundaries in the representation space. One typical example is the well known Dalal and Triggs's person detector [3], which represents images with HOG features and classifies person vs. background bounding boxes with SVM classifiers [2].

However, this type of approaches does not seem to be relevant to the detection of small targets on complex backgrounds. First, if the background is rich and the number of (positive) training images is limited, learning reliable discriminative features without over-fitting can not be done without strong regularization, which contrast with the need of having an accurate model of the targets. Second, targets and backgrounds have so different visual properties that it is hard to believe that the same models/features can be adapted to both. Based on these observations, we propose a detection algorithm using two distinct models, one for the background and another for the target, combined to score the candidate windows.

Manifolds are good candidates to model accurately small targets. If a target size is *e.g.* 40×40 pixels its visual appearance lies into a $1,600$-d space despite the fact that only a small number of parameters (among them: the pose, the illumination, etc.) are sufficient to explain its appearance. Manifolds are precisely adapted to represent high dimensional subspaces that can be generated from a space of fewer dimensions. Supporting this assumption, the work of Zhang [22] shows that images of 3D objects seen from different view points can be represented as points on a low-dimensional manifold. On the other hand, backgrounds do not require (and can not) be modeled as accurately as targets. Regarding their modeling, we follow the work of [1] and use a PCA based manifold model. Finally, target and background models are combined within a probabilistic framework.

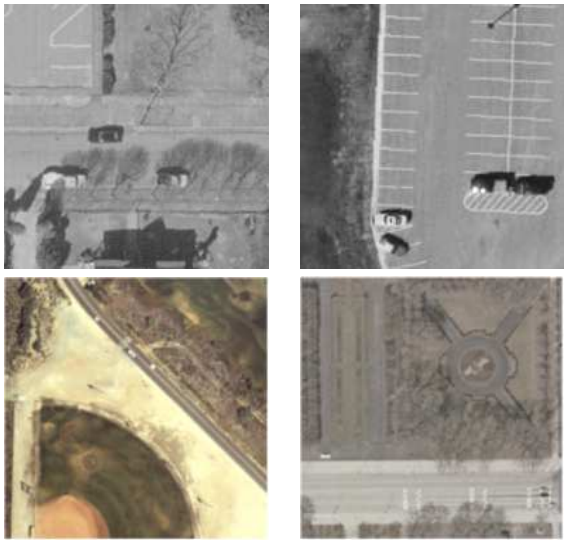


Figure 1. Typical images from the OIRDS dataset. Small size vehicles have any orientations. Shadows, highlights and complex textured background make the task very challenging.

The rest of the paper is as follows: we first introduce some related works, present our approach, and, finally, give an experimental validation on vehicle detection from aerial images, showing our approach constantly outperforms state-of-the-art object detectors.

2 Related works

Object detection is a very active area of computer vision [5]. Most of the recent approaches use the *sliding windows* framework, proposing powerful descriptors [3], new kernels [21], efficient pruning strategies [13] or new object models [6]. However, none of these recent works are really adapted to the detection of small objets.

Regarding the detection of small objects, state of the

art approaches are usually based on *saliency detection*, the objects to be detected being defined as the regions of the image which do not have the same statistics as the background [19]. Among the rare papers which tried to explicitly model the targets, we can mention the work of [15], which – in addition of introducing a new dataset of 36×18 pixels pedestrian images – have shown that good performance can be obtained by combining standard features such as Haar wavelets or HOG features with SVM/boosting classifiers [4]. [11] presents interesting vehicle detection results by using large and rich set of application-specific image descriptors. Unfortunately their experiments are not reproducible (protocols not given in the paper).

Manifold learning has already been used by several authors to address detection tasks. In [17], Pentland *et al.* introduced the well known *eigenfaces*, using Principal Component Analysis to build linear face manifolds used for face detection. It has also been applied later to hand detection in [14]. In the same spirit, [1] uses PCA for object detection, by modeling background and objects as linear manifolds. Interesting results are reported on good quality car and pedestrian images, for high dimensional manifolds. In [7], the authors used autoencoders to build face manifolds for face detection. However, this approach is limited by the lack of background model.

Our approach builds on these recent works by using the best current image features within a manifold learning framework. The contribution of the paper lies in the combination of two types of manifolds, namely autoencoders for the targets and linear manifolds for the backgrounds. As far as we know, this is the first time such a model is proposed.

3 Our approach

Our approach builds on the standard *sliding window* framework (*e.g.* [3]), which consists in classifying densely extracted image sub-windows as foreground or background regions, and applying a non-maximum suppression post processing stage. The contribution of the paper lies in the model used in the scoring function.

As said before, we use two (probabilistic) distinct models, one for backgrounds the other for objects, the score of a candidate window being computed as their log-likelihood:

$$S(X) = \log\left(\frac{p_{obj}(X_l = obj|X_s)}{p_{back}(X_l = back|X_s)}\right) \quad (1)$$

where X_l is the (unknown) class of the window ($X_l \in \{obj, back\}$), X_s is the signature of the window (*i.e.* any visual descriptor such as a HOG descriptor). Probabilities p_{obj} and p_{back} are given by the object and background models respectively. Please note that, as we have two distinct models, $p_{obj} \neq 1 - p_{back}$ (which contrasts with standard approaches using a single model).

Both classes (*i.e.* objects and background) are modeled by manifolds learned during a training stage. If $X_{s,t} \in H$ denotes the training signatures[1] representative of a class (H is the signature space), building a Riemanian manifold $\mathcal{M}$ representative of these signatures is equivalent to finding a function f, such as (following the Nash embedding theorem):

$$\forall X_{s,t} \in \mathcal{M}, \exists! Y \in \mathcal{R}^n, Y = f(X_{s,t}) \quad (2)$$

f is called the embedding of $\mathcal{M}$, and is an isometric function. Obviously, if X_s lies on the manifold, $f^{-1} \circ f(X_s) = X_s$. f^{-1} projects any point of the input space onto the manifold M. By denoting $P_{\mathcal{M}} = f^{-1} \circ f$, we can define the distance to the manifold by:

$$D_{\mathcal{M}}(X_s) = |X_s - P_{\mathcal{M}}(X_s)| \quad (3)$$

where $|y|$ represent the Euclidian norm of y. Finally, we use this distance to derive the probability for a signature X_s to be generated by the manifold $\mathcal{M}$:

$$p(X_s \in \mathcal{M}|X_s) = \alpha \exp\left(-\frac{D_{\mathcal{M}}(X_s)^2}{\sigma^2_{\mathcal{M}}}\right) \quad (4)$$

where α is a normalization factor and $\sigma^2_{\mathcal{M}}$ a parameter of the model. In practice, as scores are given by a likelihood ratio (eq. (1)) and as we are only interested in ranking candidate windows, the normalization factor can be ignored. The only reamining parameter is the object/background ratio of $\sigma^2_{\mathcal{M}}$, estimated by cross-validation.

Object manifolds. Object manifolds are given by autoencoders [12]. Indeed, in addition of being reported as being efficient for several computer vision tasks, they make the computation of f and f^{-1} possible, which is not the case of most of the manifold models (such as ISOMAP [20], or LLE [18]). Furthermore, they allow to build very expressive models whose complexity can be adapted by varying their number of layers (3 in our case) and hidden neurons (fixed by cross-validation in our experiments). We train our autoencoders by minimizing the reconstruction error of training examples; *i.e.* $Error = \sum_{X_{s,t} \in train} (X_{s,t} - g \circ f(X_{s,t}))^2$, where f is the function connecting the input to the central layer of the autoencoder, and g the function connecting the central layer to the output. $g \circ f$ is then equivalent to the previously seen $f^{-1} \circ f$. In the context of manifold learning, the network is usually used to learn f and f only, providing an embedding of the data [9]. In contrast, we keep the full network, which gives us the projection $P_{\mathcal{M}}(X_s)$ we are looking for. In practice, we use sigmoid activation functions and train autoencoders, after doing a contrastive divergence initialization [8], with a standard back-propagation algorithm. Contrastive divergence is the key to good results, as it helps the neural network to focus on data that were given (instead of the identity function). In practice, to learn the manifold, we take a representative set of training windows, compute their signatures and optimize autoencoder parameters as explained above.

Background manifold. Our hypothesis is that linear models such as the PCA is best suited to model backgrounds. A signature X_s can be written as $X_s = \sum \beta_i * PC_i$ where β is the representation of X_s in the PCA basis (PC_I are the principal component). We can then project X_s into a N-dimensional subspace using the N first principal components.

$$X_s = \sum_{i=1:N} \beta_i PC_i + \sum_{j=N+1:M} \beta_j PC_j = P(X_s) + \bar{P}(X_s) \quad (5)$$

[1] we use terms *signatures* and *visual features* indistinctively.

$P(X)$ is the projection of X on the manifold while $\bar{P}(X)$ is the projection on the space orthogonal to manifold. Interestingly, $|\bar{P}(X)|$ is the distance to the manifold, which is proportional to the mean square reconstruction error. In our experiments, we randomly sample background windows from training images, compute their signatures and find the best basis by doing a SVD decomposition of their covariance matrix.

Image features and non-maximal suppression. Our algorithm can use any type of image features. In our experiments we have used 3 different ones: (a) raw pixel intensities (b) gradient maps (c) HOG features. Regarding non-maximal suppression, we use a simple and efficient iterative strategy consisting in keeping only the windows which have the maximum score over a disk (which radius is half the window width).

4 Experimental results

Dataset and protocol. We validated our approach on the OIRDS dataset [16], which is one of the rare publicly available dataset for Automatic Vehicle Detection with aerial images.

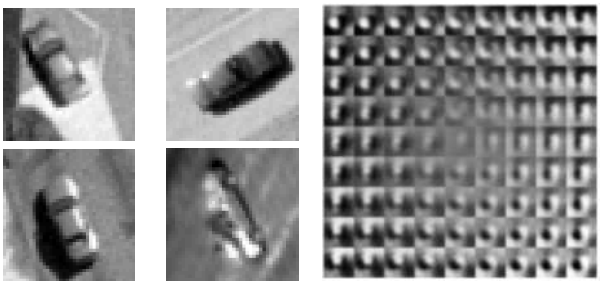
OIRDS contains a compilation of around 1,000 aerial images from different sources (*e.g.* USGS and VIVID), with about 1800 targets. Shadows, specularities, occlusions, as well as the large intra-class variability (*e.g.* regular cars, pickups, mini-vans, etc.) make this dataset very challenging. The dataset is provided with rich annotations: distance from the camera to the ground, target size (in pixels), bounding boxes, percentage of occlusion, type, etc. are given for each vehicle. Fig. 1 shows typical images from this dataset. The dataset is split in 10 folds and we evaluate the performance using a 10 fold cross validation procedure, by reporting the mean average precision (we use the experimental protocol of [5]).

As we are primarily interested in knowing the performance of our detector for small targets detection, images were downscaled to produce a dataset in which targets are not bigger than 40×40 pixels.

Finally, as (unfortunately) no reproducible results have been published so far on this dataset (nor on any publicly available dataset for small target detection) we compare the performance of our algorithm with [3], known to get state of the art results on such tasks (comparisons with the part-based model of [6] would not make sense because of target sizes). In addition, we have also implemented a generative model based on a Gaussian mixture model, which is reference for generative models.

Implementation details. Training data are obtained by cropping positive examples, which gives a total of about 3800 positive examples per fold, and 13000 negative windows randomly sampled from the background (no overlap with objects). As the step size of our sliding window is of 8 pixels, when we crop positive images for training we add a random shift up to 4 pixels to make the model more tolerant to small shifts. Some typical positive training examples are given Fig. 2(a). In addition, the training set is extended by adding positive examples obtained by flipping up/down/left and right and by rotating the initial training set.

Regarding image signatures, we experimented with three different signatures: (1) normalized raw level intensities, often used for target detection (2) image gradients, supposed to be more robust to illumination changes and (3) HOG features, considered to be the best choice for this task. In practice, raw pixel intensities are computed as the mean of the different color channels (OIRDS images are in color). Gradient images are computed by a Sobel filter. Finally, HOG31 is an histogram of oriented gradient, with 8 pixels overlap cells of 16×16 pixels. They contain a 9 bins histogram of unsigned orientations concatenated with a 18 bins histogram of signed orientations.



(a) Typical training images with shadows and specular spots. (b) Vehicle manifolds learnt by our autoencoder.

Figure 2. Sample training images and autoencoder's manifold.

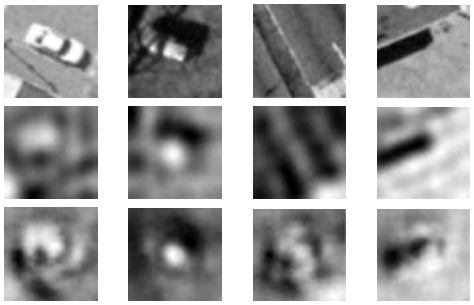


Figure 3. Four candidate windows and their reconstruction by the PCA background manifold (2nd row), and by the target manifold learnt by an autoencoder (last row).

The linear SVM classifier is taken from the *svmlight* library [10]. The Gaussian mixture models were learnt with the Expectation Maximization algorithm and include 3 gaussians components. The manifold dimensionality of backgrounds models is of 40, 10 and 16 for intensity, gradient and HOG signatures respectively. Autoencoders have 3 layers and have respectively 35, 8 and 10 inner nodes for intensity, gradient and HOG signatures. All these values were determined by preliminary experiments and were kept fixed for all the presented experiments.

Visualizing autoencoder and PCA reconstructions. Fig. 2(b) shows some vehicle appearances our autoencoder can generate once trained. The model has learnt rotated appearance of cars. Fig. 3 shows 4 candidate windows (1st row), their projection on the background manifold (obtained by PCA, 2nd row), as well as their projection on the car manifold given by an autoencoder (last row). As it can be noticed, target images are better reconstructed by the target model than by the background one, and vice-versa.

Quantitative results. We have experimented with 5 different detectors. The first one (so called AE-PCA) is ours, using an autoencoder to model targets and a PCA based manifold for backgrounds. The second is

	Intensity	Gradient	HOG31
GMM-GMM	8.3%	21,3%	17.7%
HOG-SVM [3]	10.5%	35.2%	46.8%
PCA-PCA	35.0%	37.9%	42.5%
AE-AE	35.3%	33.5%	47.5%
AE-PCA (ours)	**35.5%**	**39.9%**	**48.9%**

Table 1. Mean Average Precision on OIRDS.

	Intensity	Gradient	HOG31
HOG-SVM [3]	1.5%	12.1%	12.6%
AE-PCA (ours)	**3.3%**	**16.4%**	**17.1%**

Table 2. Mean Av. Precision on large images.

one of the state of the art approaches for detection, namely the Dalal and Triggs's detector [3] (so called HOG-SVM). In addition, we have also experimented with three other detectors, one using PCA for target and background (PCA-PCA), another one using gaussian mixture model, here again for both target and background (GM-GM), and a last one using an autoencoder for both as well (AE-AE). For these 5 detectors, we report the mean average precision over the 10 folds of the OIRDS datasets in Table 1.

The main conclusion we can draw from these results is that the proposed approach (the AE-PCA detector) outperforms any other detector, for any type of feature. The best results are obtained with HOG31 signatures. We also observe that Gaussian mixture models do not perform well in any case. Indeed, we have noticed that the GM model tends to be specialized to a few images, showing that EM gets stuck in local minima. From these results, we can also conclude that the HOG-SVM detector is outperformed – when using gradient and gray level signatures – by the PCA-PCA detector. HOG-SVM is however better than PCA-PCA with HOG31 features. In addition, we can also observe that using two autoencoders (objects+backgrounds) does not give better results, as the background autoencoder fails to capture the diversity of backgrounds.

We kept aside a dozen images that were more difficult because of their large size for additional experiments (using the previously learnt classifiers). Results are given Table 2. The performance is not as good as on the regular OIRDS images, as images are much larger while not containing more targets. Nevertheless, our AE-PCA detector clearly outperforms the HOG-SVM detector.

5 Conclusions

This paper proposes a detection algorithm based manifold learning in which targets and background are model by distinct and adapted models. The object is accurately modeled by the mean of an autoencoder learned off-line. In addition, background is modeled by a PCA based linear manifold. We have experimentally validated our approach on a publicly available vehicle dataset, and show results that outperform state-of-the-art algorithms.

Acknowledgments. This work was supported by ANRT through the CIFRE sponsorship No 2011/0850 and by SAGEM-SAFRAN group.

References

[1] G.V. Carvalho, L.B. Moraes, J.D.C Cavalcanti, and I.R. Tsang. A weighted image reconstruction based on pca for pedestrian detection. In *IJCNN*, 2011.

[2] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3), 1995.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[4] M. Enzweiler and D.M. Gavrila. Monocular pedestrian detection: Survey and exp. *PAMI*, 31(12), 2009.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal voc challenge. *IJCV*, 88(2), 2010.

[6] P. Felzenszwalb, R. Girshick, D. Mcallester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9), 2009.

[7] R. Feraud, O.J. Bernier, J.E. Viallet, and M. Collobert. A fast and accurate face detector based on neural networks. *PAMI*, 23, 2001.

[8] G. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14:2002, 2000.

[9] G. E. Hinton and R. R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313:504–507, July 2006.

[10] T. Joachims. Making large-scale support vector machine learning practical, 1999.

[11] A. Kembhavi, D. Harwood, and L.S. Davis. Vehicle detection using partial least squares. *IEEE PAMI*, 33(6), 2011.

[12] M.A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2), 1991.

[13] C.H. Lampert, M.B. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. In *CVPR*, 2008.

[14] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *PAMI*, 19(7), 1997.

[15] S. Munder. An experiment study on pedestrian classification. *PAMI*, 28(11), 2006.

[16] F. Tanner B. Colder C. Pullen D. Heagy M. Eppolito V. Carlan C. Oertel and P. Sallee. Overhead imagery research data set: an annotated data library and tools to aid in the development of computer vision algorithms. In *Proc. of IEEE Applied Imagery Pattern Recognition Workshop*, 2009.

[17] A. Pentland. Viewbased and modular eigenspaces for face recognition. *CVPR*, 1994.

[18] L.K. Saul and S.T. Roweis. Think globally, fit locally: unsupervised learning of low dimensional manifolds. *J. Mach. Learn. Res.*, 4, 2003.

[19] H.J. Seo and P. Milanfar. Visual saliency for automatic target detection, boundary detection, and image quality assessment. In *ICASSP*, 2010.

[20] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290, 2000.

[21] A. Vedaldi and A. Zisserman. Sparse kernel approximations for efficient classification and detection. In *CVPR*, 2012.

[22] X. Zhang, X. Gao, and T. Caelli. Parametric manifold of an object under different viewing directions. In *ECCV*, 2012.