Model selection in pedestrian detection using multiple kernel learning

Frédéric Suard, Alain Rakotomamonjy, Abdelaziz Bensrhair
LITIS EA4051, INSA/Univ de Rouen
avenue de l'université, 76800 Saint Etienne du Rouvray, France
email: *frederic.suard@insa-rouen.fr*

***Abstract*— This paper presents a pedestrian detection method based on the multiple kernel framework. This approach enables us to select and combine different kinds of image representations. The combination is done through a linear combination of kernels, weighted according to the relevance of kernels. After having presented some descriptors and detailed the multiple kernel framework, we propose three different applications concerning combination of representations, automatic parameters setting and feature selection. We then show that the MKL framework enable us to apply a model selection and improve the performance.**

I. Introduction

Since many years now, pedestrian detection from images has been source of many researches. This topic of research is usually decomposed in two parts : the first one consists in searching for discriminative features while the second part deals with the learning of a decision function from these features.

Recent works [5], [6], [2] have been proposed for representing pedestrian images. For instance, Papageorgiou et al. have used a wavelet decomposition approach for extracting features from images while Shashua et al. have considered histograms of gradients. For both approaches, the underlying objective is to extract from images some discriminative features that help a classifier to recognize images containing or not a pedestrian. These are two examples among the many recent researches that have dealt with the construction of different features.

Once some features have been extracted, the second stage of an automated pedestrian detection algorithm resides in the pedestrian classification problem. During the last years, kernels methods, like Support Vector Machines [9] have shown their efficiency to address such problems [2], [8], [4].

When using SVMs or any other kernel methods, features are integrated into the classifier through a kernel function $\mathbf{k}(\mathbf{x}, \mathbf{x}')$, where $\mathbf{x}$ and $\mathbf{x}'$ represent the features from two different images. In such context, the kernel function acts, in some way, as a measure of similarity between features $\mathbf{x}$ and $\mathbf{x}'$ which can be non-vectorial representations.

Recently, a SVM algorithm using multiple kernels have been introduced [3]. The underlying idea of such algorithm is the combination of different heterogenous source of information for learning a decision function. Indeed, this multiple kernel learning (MKL) framework defines the kernel function as a linear combination of kernels :

$$\mathbf{k}(\mathbf{x}, \mathbf{x}') = \sum_{k=1}^{K} \beta_k \mathbf{k}_k(\mathbf{x}, \mathbf{x}')$$

where each kernel has been computed from one specific representation of the data. From this equation, it is clear that the idea consists in defining a set of kernels and in combining and selecting a subset of these kernels, each β being the weight of a single kernel in the overall combination.

One advantage of the MKL approach resides in the possibility to combine and select the most relevant representations. In fact, the set of kernels can be composed of different kernels built from different types of representations. So instead of using a single representation, one can fuse different representations through the kernel.

One drawback of kernel methods is the need of tuning efficiently the classifier and kernel parameters. Thanks to multiple kernel, it is possible to address this issue by proposing a set of kernels, each of them being computed with a different set of kernel parameters.

Another advantage of multiple kernel is the possibility of selecting only a subset of features. This approach could be assimilated as feature selection, since we can build a set of kernels from each feature and then selecting only the most relevant one by means of the MKL algorithm.

In this paper, we propose to analyze the contribution of multiple kernel learning framework for pedestrian detection. At first, we propose to present the multiple kernel framework and detail the theoretical advantages of such approach. After having highlighted this approach, we propose to combine different kinds of features for classifying pedestrian images. For this purpose, we will present different classical features used in the literature for describing pedestrian and will use multiple kernel approach for combining and retaining the most relevant feature. By doing so, we expect to the algorithm to have a better classification performance due to the feature combination.

We also investigate the contribution of multiple kernel for selecting automatically the kernel parameter and for selecting features.

This paper is organized as follows. In section II, we present the different features that have been extracted from pedestrian images and that have been used afterwards for kernel combination. Then, in section III we detail the multiple kernel learning framework. Finally, the section IV presents our results for pedestrian detection using multiple kernel learning for feature combination, kernel parameter selection and feature selection.

II. Image representation

In this section we will describe some methods for characterizing an image. The aim consists in finding some relevant information to describe the content of images.

In our case, we suppose that our images contain only one pedestrian, which is centered. All images have the same size, in our case 128×64 pixels.

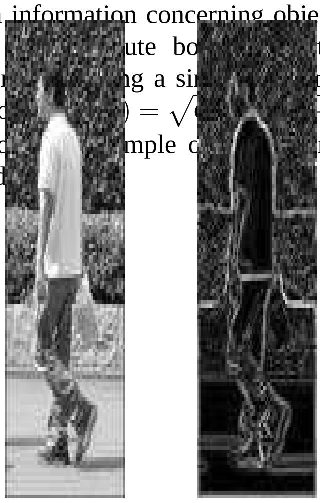
A. Pixel value

The first feature is based on the original value of each image. This descriptor can be considered as a reference, since we applied no transformation to images. Xu et al. [10] employed this method to characterize image with infrared images. Figure 1 left shows an example of pedestrian image.

B. Gradient norm

The second descriptor uses the value of gradient norm. We can then retain information concerning object edges present in the image. ... ute bo... tal and vertical gradient G_H a... g a si... ng $[-1\ 0\ 1]$ and compute the no... $) = \sqrt{\ldots + G_V(x,y)^2}$.

Figure 1 sho... mple o... n image and its associated grad...



Original image   Gradient norm

Fig. 1. Pedestrian image (left) and its gradient norm (right)

C. Wavelet

A descriptor based on wavelet has been proposed by Papageorgiou et al. [5] for pedestrian detection. In this paper, we have used as a feature a similar approach. The aim is to apply a Haar wavelet transform at a scale n, that is to say to transform a neighboring of size 2^n pixels. The distance between two neighboring is $\frac{1}{4}2^n$ pixels. For each neighboring, we apply a vertical, horizontal and diagonal Haar wavelet and add all coefficients obtained during the transformation to a single vector. This vector is then considered as the image descriptor.

D. Histo...

We pr... of orient... tors resi... are split...

The fi... been pro... in many... each reg... All histo... vector w...

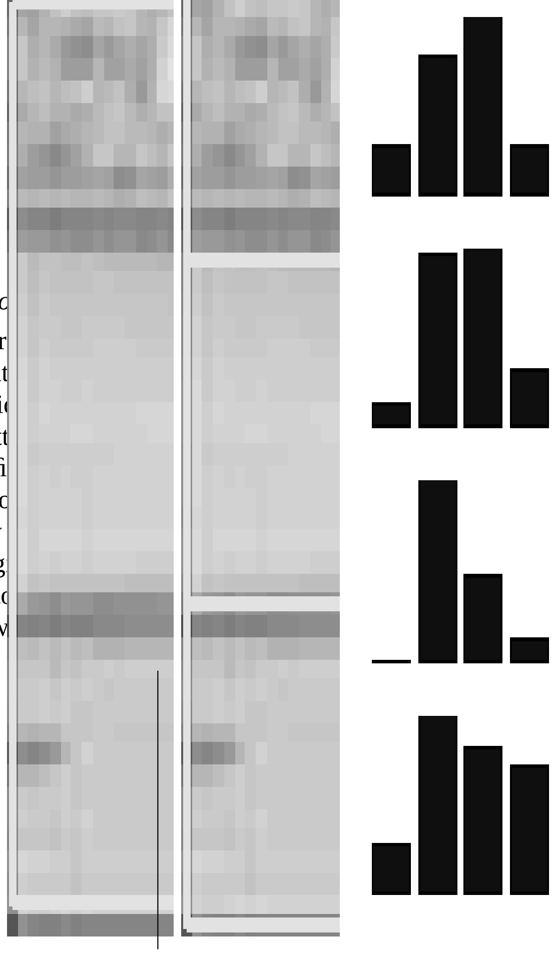


Fig. 2. Image splitting according to Shashua method (left), image splitting accorgind to Dalal method (middle) and example of histograms obtained for this image (right).

The second descriptor, that we will called HogDalal, has been introduced by Dalal et al. [2] and uses also local histograms of oriented gradients. On the contrary to the method proposed by Shashua, the image is cutted regularly and does not considering the pedestrian morphology.

To obtain a descriptor, the procedure is the following:

1) compute both norm and orientation of the gradient,
2) split image into cells,
3) compute one histogram for each cell (look at the right image on figure 2),
4) normalize all histograms within a block of cell.

The final descriptor is obtained by adding all normalized histograms into a single vector. This descriptor has been much more detailed in [2], [8].

III. Multiple Kernel

The learning algorithm we use in this paper is a Support Vector Machines classifier. The SVM classifier is a binary classifier, based on supervised learning, that looks for an optimal hyperplane as a decision function in a high-dimensional space [9]. Thus, consider one has a training data set $\{\mathbf{x}_k, y_k\} \in \mathcal{X} \times \{-1, 1\}$ where $\mathbf{x}_k$ are the training examples vector and y_k the class label, for $k = 1 : N$.

The decision function is of the form :

$$f(x) = \text{sign}\left(\sum_{i=1}^{N} \alpha_i y_i \mathbf{k}(\mathbf{x}_i, \mathbf{x}) + b\right) \quad (1)$$

with $\mathbf{k}(.,.)$ a kernel, α and b some variables learned from the training set.

Recently Lanckriet et al. [3], have shown that using multiple kernel learning instead of a single kernel can improve

Fig. 3. Examples of pedestrians (first line) and non-pedestrians (second line) manually extracted.

the classifier performance. They stated that the gain in performance can be considerable if different kernels have been obtained from heterogenous sources of information.

The main idea underlying MKL is to consider a new kernel $\mathbf{k}$ as a convex linear combination of other kernels :

$$\mathbf{k}(\mathbf{x}, \mathbf{x}') = \sum_{k=1}^{K} \beta_k \mathbf{k}_k(\mathbf{x}, \mathbf{x}') \quad (2)$$

with $\beta_k \geq 0$, $\sum_k \beta_k = 1$ and $\mathbf{k}_k(.,.)$ a kernel using a single representation using all features composing x or a subset of features. All kernels $\mathbf{k}_k$ can also involve different kernel functions such as polynomial or Gaussian kernel using different parameters. Within this framework, the problem of data representation is transferred to the choice of β_k.

The value of coefficients α, b and β are obtained by solving the dual of the following optimization problem:

$$\begin{cases} \min_{\mathbf{w},\beta,b,\xi} & \frac{1}{2}\left(\sum_{k=1}^{K} \beta_k \|\mathbf{w}_k\|_2\right) + C\sum_{i=1}^{N}\xi_i \\ \text{s.t.} & y_i f(\mathbf{x}_i) \geq 1 - \xi_i \quad \forall i = 1, \dots, N \\ \text{and} & \sum_{k=1}^{K} \beta_k = 1 \end{cases} \quad (3)$$

Bach et al. [1] derived this dual formulation and wrote equivalently :

$$\begin{cases} \max_{\gamma,\alpha} & \gamma \\ \text{s.t.} & \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j \mathbf{k}_k(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^{N} \alpha_i \leq \gamma \\ \text{and} & \sum_{i=1}^{N} \alpha_i y_i = 0, \quad \forall i, 0 \leq \alpha_i \leq C \end{cases} \quad (4)$$

Hence, finding the optimal solution can be done by solving the following semi-infinite linear program (SILP). We use the formulation of Sonnenburg et al. [7]:

$$\begin{cases} \max_{\theta,\beta} & \theta \\ \text{s.t.} & \sum_{k=1}^{K} \beta_k = 1 \\ \text{and} & \sum_{k=1}^{K} \beta_k S_k(\boldsymbol{\alpha}) \geq \theta \end{cases} \quad (5)$$

with $S_k(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j \mathbf{k}_k(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^{N} \alpha_i$. Sonnenburg et al. proposes to use the Column Generation technique to solve this problem. The idea of the algorithm is to find the optimal value of β and θ for a subset of constraints and determine if $\boldsymbol{\alpha}$ satisfies the constraints $\sum_{k=1}^{K} \beta_k S_k(\boldsymbol{\alpha}) \geq \theta$. If the constraints are satisfied then the solution is optimal otherwise some constraints are added to the constraints set and the process is continued until convergence of the β values is achieved. For each step, a standard SVM solver is used for processing $S_k(\boldsymbol{\alpha})$ with an update of the kernel since some constraints and value of β can change.

For this paper, we used an optimized version of the algorithm proposed by Sonnenburg et al. [7]. A Matlab code of our implementation is available on demand.

IV. Results

In this section we will present some results. We then show the interest of the multiple kernel framework approach for combining images representations, for choosing the optimal parameters and for selecting feature.

For this work, we took 310 urban scenes from which we manually extracted 1240 pedestrians and 6220 non-pedestrians. Images were captured during different weather and ligthning conditions : night and day, cloudy and raining weather. Pedestrian are centered and can be partially occluded. Some examples of pedestrians and non-pedestrians extracted are shown on figure 3. All images extracted are rescaled to the same size : 128×64 pixels.

To compare the results, we plot the rate of true positive against the rate of false positive by counting the number of correct detections and false alarms, when a threshold putted on the prediction value of $f(x)$ (see eq. 1) varies. The AUC is the value of the area under the ROC curve, which should be the nearest of 1 for the best result.

A. Combining representations

This test consists in combining different representations. We first extracted a set of descriptors for each image of the learning set as presented in section II.

For each type of descriptor, we computed a linear kernel and added all to a set of kernels. We applied the MKL algorithm and retained the most relevant representations.

This last point is simply done by considering the value of coefficients β_k (see eq. 2). When a coefficient is null, the associated kernel has no influence in the solution, so that the representation is not relevant. On the contrary, when a coefficient value is up to 1, the representation is very relevant.

We trained the classifier with a learning set containing 500 pedestrians and 500 non-pedestrians, and evaluated this

Fig. 4. This figure shows the variation of β for each kernel computed from various representations.

On figure 4 we shows the value of β for each representation for each iteration. We can note that the β corresponding to the representation of HogDalal obtained the largest value, which means that this representation is very relevant compared with other. We also can note that globally the value of β does not vary if we consider all test.

The average value of each β is then computed :

Kernel	Pixel	Gradient	Wavelet	HogShashua	HogDalal
β	0.0453	0.0571	0.0029	0	0.8947
variance	0.0115	0.0215	0.0090	0	0.0211

We can note that the kernel coming from the HogDalal descriptor has a large weight in the final kernel.

Parrallely, we have improved the performance of each representation separately. The following table shows the average value of AUC obtained.

Method	Pixel	Gradient	Wavelet	HogShashua	HogDalal
AUC	0.8806	0.8963	0.7623	0.9592	0.9827
%	0.8259	0.8171	0.7237	0.8919	0.9399

Using the MKL approach, we obtained an AUC of 0.9859 and a good recognition rate of 0.9472%, which performs better than any single representations. We can also notice that some descriptors are more relevant than other, in particular, using histograms of oriented gradient with a cutting proposed by Dalal reveals to be very efficient,

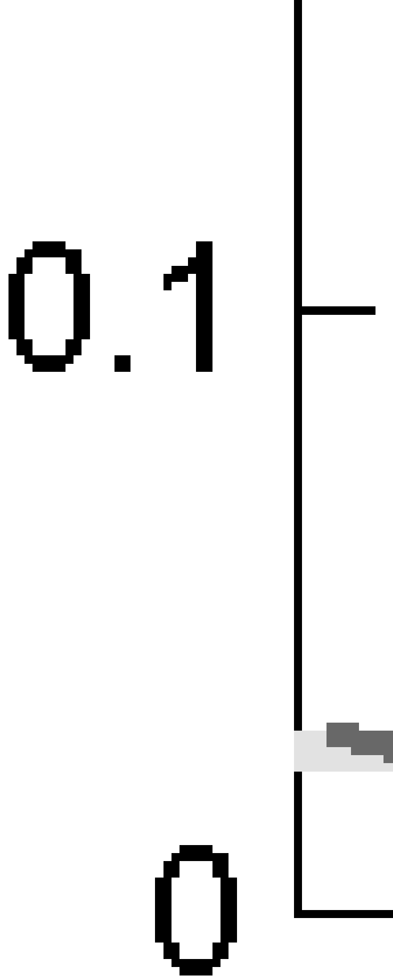


Fig. 5. This figure shows the variation of β for each kernel computed from various representations.

We also plot all values of β obtained for each test on figure 5. We can see that the value of most relevant presentation is lower than the value of the coefficient HOGDalal. Moreover, the variance for all coefficients is larger, since all kernels participate to the solution.

We can notice a relation between the HogShashua and the Gradient information. This relation can be explain by the fact that these descriptors involve the same type of information but the HogShashua exploits only a subpart of the image. When all the pedestrians are well centered this descriptors reveals to be more efficient. On the contrary, the gradient information is more relevant.

So we can conclude that the MKL approach can select the most relevant representation, but is also capable to avoid the redundancy of information. A combination of representations also improves the global performance.

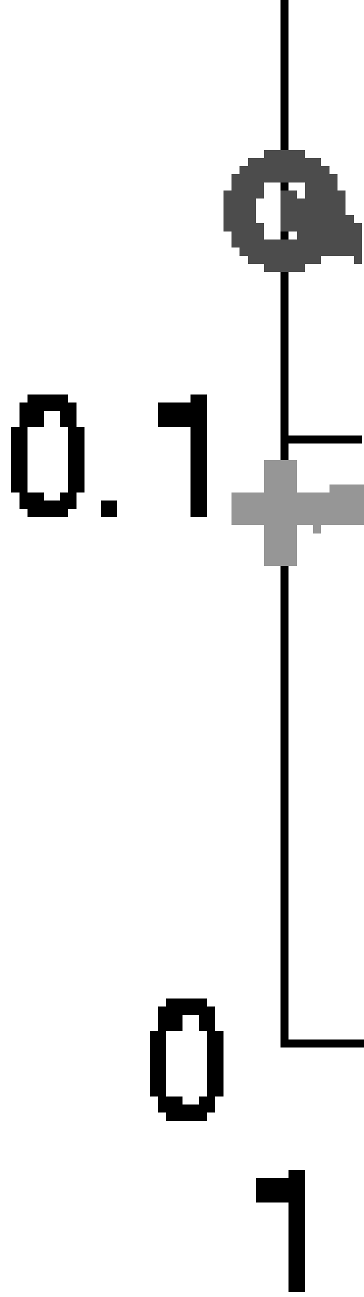


Fig. 6. This figure shows the variation of β for each kernel computed with the HOG descriptor, when the parameters of the kernels and the parameter C vary.

Figure 6 shows the average value of each β when C varies. We can note that the value of β depends on the C value. When C is lower than 1, the main kernel is based on a polynomial kernel, but when C is greater than 1, the main kernel is a gaussian kernel with a bandwidth of 1. It can be explain by the fact that this parameter balances the weight between the regulariztion of the solution and the performance. With a higher value of C, the solution is more complex but we tolerate less errors.

We can also notice that, when $C \geq 1$, the combination involves differents kernels : a polynomial kernel and 4 gaussian kernels. This combination performed better compared to $C < 1$, when the MKL retained only the polynomial kernel.

For $C = 1$, we obtained the following results comparing the value obtained for the MKL and each kernel separately:

Kernel	**AUC**	**%**
MKL	0.9645	0.8976
Linear	0.9580	0.8758
Gaussian, $\sigma = 0.1$	0.8926	0.8520
Gaussian, $\sigma = 0.5$	0.8937	0.8432
Gaussian, $\sigma = 1$	0.9188	0.8646
Gaussian, $\sigma = 2$	0.9499	0.8888
Gaussian, $\sigma = 5$	0.9598	0.8438
Gaussian, $\sigma = 10$	0.9580	0.5972
Gaussian, $\sigma = 20$	0.9515	0.5120
Gaussian, $\sigma = 50$	0.8664	0.5682

This table shows that we can improve the global performance by combining different types of kernels.

C. *Feature selection*

The last test can be considered as feature selection problem. For this experience, we are using only the HogDalal descriptor. One specificity of this descriptor is to split the image into several cells. For each cell, one kernel is computed and added to the kernel set. The aim of this test is then to use multiple kernels in order to retain the most relevant cells by considering the value of coefficients β_k (see eq. 2).

To build the kernel set, we computed one kernel for each cell using a gaussian kernel with a bandwidth of 1, which is the best kernel retained during the previous test. The learning and test set contained 500 pedestrians and 500 non-pedestrians. Each set was randomly constitued 20 times for each iteration. The weight for misclassified point (parameter C in eq. 3) is fixed at 1.

We tried different configurations for the parameter set :

Parameter set	A	B	C	D	E	F
size of cell (pixels)	8	8	16	16	32	32
Size of Block (cells)	1	2	1	2	1	2
Overlapping (cells)	0	1	0	1	0	1
number of kernels	128	420	32	84	8	12

For each set, histograms have 4 bins and the vote is weighted by the gradient magnitude.

We could then compare results with a feature selection strategy using MKL and a standard kernel built with all features. We set this kernel with same parameters of the MKL kernel set : a gaussian kernel with a bandwidth of 1.

	MKL			no feat. sel.	
	AUC	%	$\frac{\#(\beta > 0)}{nbkernel}$	AUC	%
A	0.9726	0.9132	0.7891	0.9105	0.7327
B	0.9810	0.9298	0.7469	0.9107	0.7272
C	0.9622	0.8917	0.8203	0.9511	0.8804
D	0.9716	0.9095	0.7679	0.9313	0.8565
E	0.9479	0.8751	0.8375	0.9564	0.8891
F	0.9277	0.8499	0.8333	0.9472	0.8819

We ca[illegible]
the numb[illegible]
can also [illegible]
few cells [illegible]
no gain [illegible]
much mo[illegible]
In this ca[illegible]

Figure [illegible]
regard of [illegible]
and C. V[illegible]
cell is ar[illegible]
corners v[illegible]

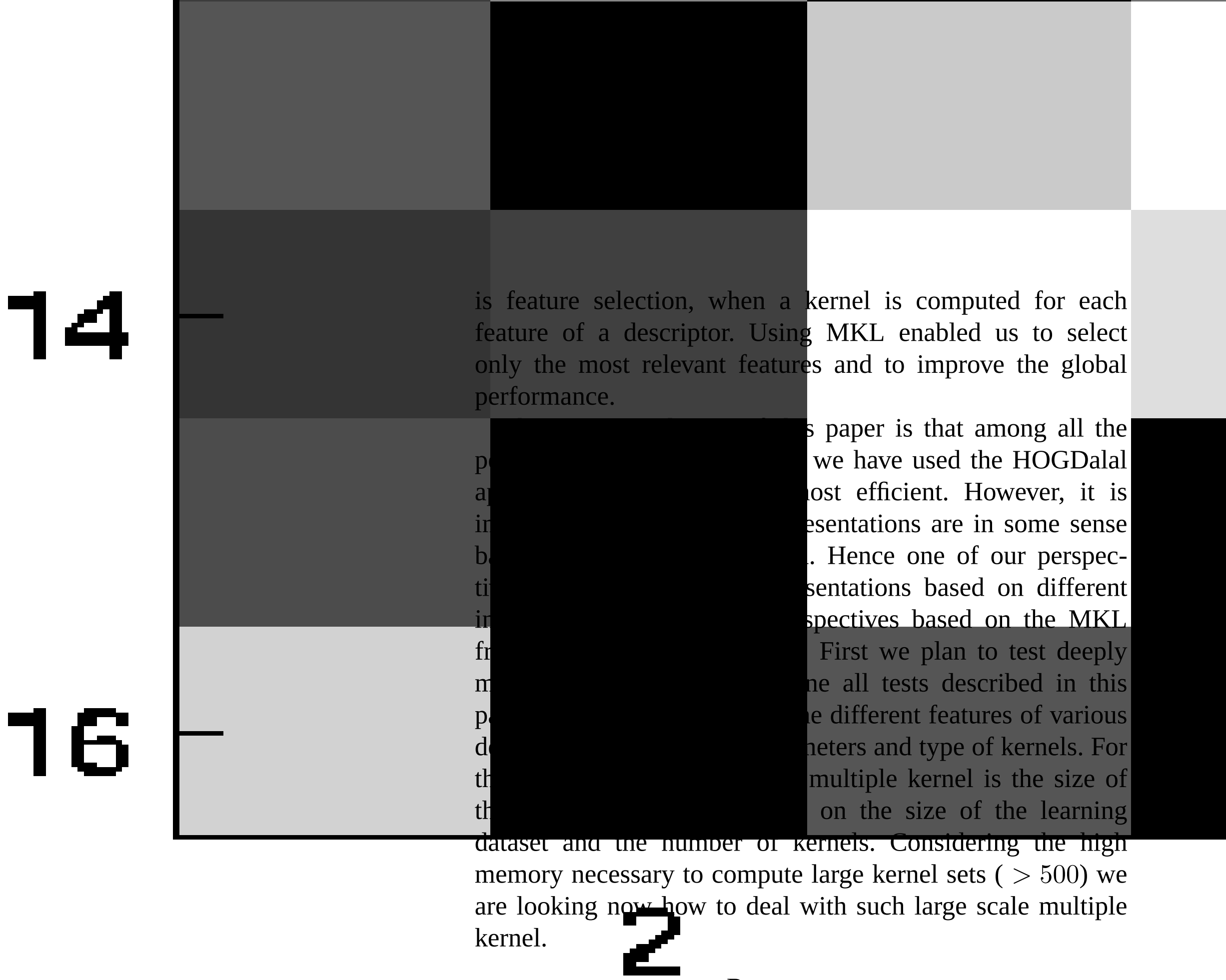


Fig. 7. This figure shows the value of β, each kernel of the kernel set corresponding to a single cell of the image for parameter set A (left) and C (right).

A last advantage of using a MKL approach to select the most relevant features resides in the complexity of the process. Classical techniques, like forward-backward, are suppose to iterate the same processus a large number of time to select the most relevant features. We have to compute all data and the associated kernel to remove or add only 1 feature at a time. With a MKL approach all we have to do is to build one kernel per feature and the algorithm will note the relevance of each kernel by means of the coefficient β.

V. Conclusion and perspectives

We have presented a novel approach for model selection using multiple kernel. The idea consists in defining a new kernel as a linear combination of various kernels. The combination is weighted by some coefficients with regard of the kernel relevance. After having introduced some standard descriptors used for pedestrian detection, we have presented in detail the multiple kernel framework. Then, we have proposed to apply multiple kernel for pedestrian detection with three different applications.

First we used MKL to combine and select different representations of images, each kernel of the set corresponding to a single representation. This approach enable us to select the more relevant application and improved the detection performance. The second application was due to automatic setting. One drawback of classical kernel machine resides in the kernel setting. Thanks to multiple kernel, we showed that it is possible to set automatically the kernel, by computing a set of kernels with different paramters. The last point is feature selection, when a kernel is computed for each feature of a descriptor. Using MKL enabled us to select only the most relevant features and to improve the global performance.

[illegible]s paper is that among all the p[illegible] we have used the HOGDalal a[illegible]ost efficient. However, it is i[illegible]esentations are in some sense b[illegible]. Hence one of our perspec- ti[illegible]sentations based on different i[illegible]spectives based on the MKL fr[illegible] First we plan to test deeply m[illegible]ne all tests described in this p[illegible]e different features of various d[illegible]neters and type of kernels. For th[illegible] multiple kernel is the size of th[illegible] on the size of the learning dataset and the number of kernels. Considering the high memory necessary to compute large kernel sets (> 500) we are looking now how to deal with such large scale multiple kernel.

References

[1] Francis R. Bach, Gert R. G. Lanckriet, and Michael I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, page 6, 2004.

[2] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In Cordelia Schmid, Stefano Soatto, and Carlo Tomasi, editors, *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 886–893, June 2005.

[3] Gert R. G. Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I. Jordan. Learning the kernel matrix with semidefinite programming. *J. Mach. Learn. Res.*, 5:27–72, 2004.

[4] Stefan Munder and Dariu Gavrila. An experimental study on pedestrian classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28–11:1863–1868, 2006.

[5] Constantine Papageorgiou and Tomaso Poggio. Trainable pedestrian detection. In *Proceedings of the 1999 International Conference on Image Processing*, pages 35–39, 1999.

[6] Amnon Shashua, Yoram Gdalyahu, and Gaby Hayon. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *Proceedings of IEEE Intelligent Vehicles Symposium*, 2004.

[7] Sren Sonnenburg, Gunnar Raetsch, and Christin Schaefer. A general and efficient multiple kernel learning algorithm. In *Advances in Neural Information Processing Systems 18*, pages 1273–1280, 2005.

[8] Frédéric Suard, Alain Rakotomamonjy, Abdelaziz Bensrhair, and Alberto Broggi. Pedestrian detection using infrared images and histograms of oriented gradients. In *Intelligent Vehicles Symposium, Tokyo, Japan*, pages 206–212, June 2006.

[9] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer, N.Y, 1995.

[10] Fengliang Xu, Xia Liu, and Kikuo Fujimura. Pedestrian detection and tracking with night vision. *IEEE Transactions on Intelligent Transportation Systems*, 6–1:63–71, march 2005.