Pedestrian Detection Using Multi-Objective Optimization

Pablo Negri[1,2(✉)]

[1] CONICET, Av. Rivadavia 1917, Buenos Aires, Argentina
pnegri@uade.edu.ar
[2] INTEC-UADE, Lima 717, Buenos Aires, Argentina

Abstract. Pedestrian detection on urban video sequences challenges classification systems because of the presence of cluttered backgrounds which drop their performances. This article proposes a Multi-Objective Optimization (MOO) technique reducing this limitation. It trains a pool of cascades of boosted classifiers using different positive datasets. A Pareto Front is obtained from the locally non-dominated operational points of the Receptive Objective Curve (ROC) of those classifiers. Using information about the dynamic of the scene, different pairs of operational points from the Pareto Front are employed to improve the performance of the system. Results on a real sequences outperform traditional detector systems.

Keywords: Multi-Objective Optimization · Pedestrian detection

1 Introduction

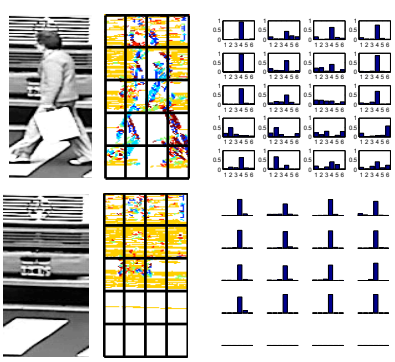
The behavior of object detection systems using image processing is controlled by fixing constrains or establishing performance criterias. Two numerical variables which can define this behavior are: Correct Detections (CD) and False Alarms (FA). CD computes objects successfully identified on the image, while FA are the erroneous outputs of the detector.

In some applications the CD ratio would be very important to identify an object or situation. For example, a buried land mines system detector should be very sensible to CD and would validate a position if there exists a slight doubt. Considering that a non detected land mine can take away a human life, a great number of FA is not relevant. On the other hand, an herbicide system using vision which has high FA ratio imply an economic waste when it fumigates unnecessarily the farmland. Minimizing the number of FA implies that not all the weeds would be eliminated. Even though, a low quantity of weed does not represent a danger for the crop. Thus, finding a balance between CD and FA will define the behavior of the system which is closely related with the application.

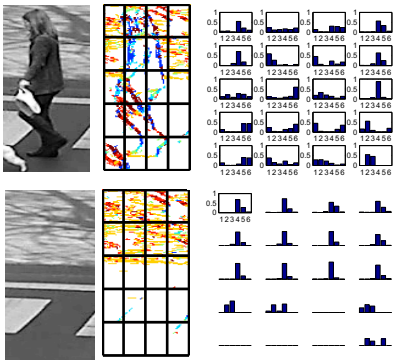
This article addresses a people detector using the Movement Feature Space (MFS) [8,9] on video sequences captured at a street corner, as show fig. 1(a). These kind of outdoors images with non controlled environments have numerous

A. Pardo and J. Kittler (Eds.): CIARP 2015, LNCS 9423, pp. 776–784, 2015.
DOI: 10.1007/978-3-319-25751-8_93

(a) Street Sequence



(b) Pedestrian 1



(c) Pedestrian 2

Fig. 1. Fig. (a) shows a capture of the street sequence. Figs. (b) and (c) represent non detected pedestrians and the features obtained from the MFS.

factors harming the performance of state of the art pedestrians detectors [4,5]. This is mainly due to: other moving objects on the scene, abrupt changes of the illumination, and a cluttered background.

Figures 1(b) and 1(c) show two examples of non detected pedestrians using MFS classifier. Second column represents the orientations matrix O_t of the MFS, and the third column shows the Histograms of Level Lines (HO2L) computed inside each patch of the grid (see [7,8] for details).

The HO2L features of the background without the Pedestrian 1 on fig. 1(b) are mostly composed of level lines with horizontal orientation: $bin = 4$. When the person is in front of the vehicle, their presence changes the histograms, but there is a high predominance of the horizontal orientation. It can be considered that their features are absorbed by the background features. Then, they are hardly noticeable by the classifier which would not detect the pedestrian. Pedestrian 2, fig. 1(c), shows a cluttered background generated by the shadows of the trees, producing moving level lines on the MFS until they became part of the background model. During this period, pedestrians walking in front of this background are not detected. The immersion effect is similar to the example 1, but less noticeable.

It can be stated that the problem is related with the presence of horizontal features. Actually, a person hardly generates this kind of features [5]. Therefore, on the training stage of a pedestrian classifier, this orientation becomes a

discriminant factor that easily eliminates FA. However, when a person is submerged on this type of background, it is not detected. To minimize this limitation, a classifier can be trained using a greater number of persons with cluttered backgrounds. But this will result in an exponential increase in the number of the FA: the horizontal level lines are not as discriminating as before. The problem has opposed objectives, and is necessary to seek for a different kind of solution.

This article proposes a technique of Multi-Objective Optimization (MOO) to minimize the effect of the horizontal features on the pedestrian detectors at the street corner. The methodology consists to train a pool of classifiers using different datasets. Their performances projected on Receiver Operating Characteristics (ROC) curves will define a Pareto Front with the operational points locally optimums [3]. Recent works of the literature apply MOO to compare performance of different algorithms [2], or obtain pools of classifiers [3,6,10,11] to choose the better combination of the training hyperparameters or features. This article, on the other hand, trains a pool of classifiers using different positive datasets to optimize the behavior of the overall detection system.

The following sections details the methodology for the training and the way the pool of classifiers in obtained. Results of the detection system are presented on section 3, finishing with the conclusions of the work 4.

2 Methodology

2.1 ROC Front Construction

Pareto optimization gives a framework where solutions of the problem coexist with opposed objectives. This pool of solutions Ψ are the different classifiers trained by our system. The vector solution $\mathbf{x} = [x_1, ..., x_n]^T \in \Re^n$ is composed of all the decision variables x_i. The l objective functions are defined as $f_i(\mathbf{x}), i = 1, ..., l$. Then, the solution $\mathbf{x}^i$ domminates solution $\mathbf{x}^2$ ($\mathbf{x}^1 \leq \mathbf{x}^2$), if and only if $\mathbf{x}^1$ is better than $\mathbf{x}^2$ on one objective and is not worst on the others [11]:

$$\forall i : f_i(\mathbf{x}^1) \leq f_i(\mathbf{x}^2) \wedge \exists j : f_j(\mathbf{x}^1) < f_j(\mathbf{x}^2) \tag{1}$$

Using this dominance concept, the purpose of the MOO algorithm consists to find the set of all dominant solutions applying the objective functions to the system. This set is denominated Pareto Front.

This work relates the objective functions $f_j(\mathbf{x}^1)$ to the ratio CD/FA obtained from the ROC curve [3]. The ROC curve is generally employed to choose an operational point for a classifier [1]. To obtain the ROC for a two class problem, a classifier is applied on a dataset composed of positives and negatives samples using different validation thresholds. The use of each threshold would result on a CD and FA point which is employed to construct the ROC curve. This curve evaluates the sensibility and specificity of the classifier.

Fig. 2(b) represents and example of two ROC curves belonging to different classifiers. The objective functions f look for CD maximization and FA minimization points. Curve ROC1 locally dominates ROC2 for high values of CD,

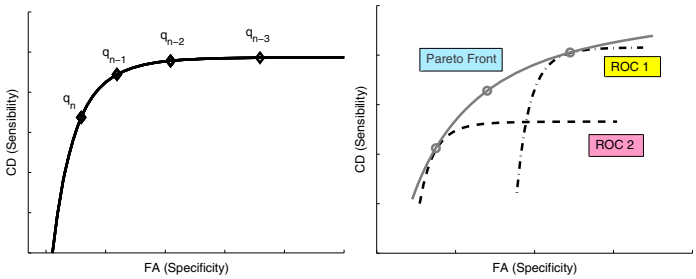


Fig. 2. Figure shows: (a) ROC curve for a Cascade of Boosted Classifiers and their operational points q_i, (b) Pareto Front using two ROC curves.

while ROC2 dominates ROC1 for low values of CD. The selection of locally dominant operational points defines the Pareto Front, which is draw on fig. 2(a) as the exterior bounding curve.

2.2 Classifiers Training

Cascade of Boosted Classifiers. This section details the methodology to combine the training of the Cascade of Boosted Classifiers, and the Multi-Objective Optimization technique. The training of a Cascade of Boosted Classifiers $\mathbf{C}$ requires a dataset P composed of positive samples of the class (pedestrians in this case), and a negative dataset N composed of non-class samples. It is also necessary to define some training parameters as: the maximum number of stages E in $\mathbf{C}$, the minimum percentage of correct detections d_{min} and the maximum percentage of false alarms f_{max} allowed at each stage [12].

The resulting Cascade $\mathbf{C} = \{C_1, C_2, ..., C_n\}$ is a set of n boosted classifiers C_i of growing complexity. Those C_i are applied sequentially on an test image to detect the positive class. The behavior of $\mathbf{C}$ is strongly related by the choose of all the training parameters and the datasets: $\{P, N, E, d_{min}, f_{max}\}$.

The ROC curve of $\mathbf{C}$ is computed employing the methodology proposed by Viola & Jones [12], considering the individual thresholds T_i of each stage C_i obtained on the training. The ROC curve is composed of the concatenation of the ROC segments calculated for each stage of the cascade. A segment j of the curve, corresponding to the C_j classifier of $\mathbf{C}$, results by applying a validation threshold to the dataset from $-\infty$ to the T_j value. Fig. 2(a) draws an example where q_n denotes the operational point of the last classifier C_n using T_n as value for their threshold, q_{n-1} corresponding to the C_{n-1} classifier, and so on.

Iterative Selection of *P*. The behavior of $\mathbf{C}$ will be strongly related by the positive samples populating the training set P. If the dataset P is highly homogeneous, through the training of $\mathbf{C}$ the positive samples projected on the classification space are easily grouped on kernels. Dissimilar samples of the mean

Algorithm 1. Multi-Objective Training

Require: P positive dataset, N negative dataset, and $\{E, d_{min}, f_{max}\}$
Ensure: Pool of Multi-Objective Cascades $\mathcal{C}_{MOO}$
1: $k \leftarrow 1$
2: $P_k \leftarrow P$, $p \leftarrow$ # positives in P
3: **while** $k \leq \mathcal{N}$ **do**
4: $\quad \mathbf{C}_k \leftarrow \text{TrainCascade}(P_k, N, E, d_{min}, f_{max})$
5: $\quad n \leftarrow$ # of stages in $(\mathbf{C}_k)$
6: $\quad s, idx \leftarrow \text{ComputeOutputScores}(\mathbf{C}_k, P_k)$
7: $\quad oidx \leftarrow \text{SortIndexIncreassingOrder}(s, idx)$
8: $\quad$ Update p: $p \leftarrow ((d_{min})^n \cdot p)$
9: $\quad$ Create P_{new} as the first p elements of P_k sorted by $oidx$
10: $\quad$ Save $\mathbf{C}_k$ in $\mathcal{C}_{MOO}$
11: $\quad k \leftarrow k + 1$
12: $\quad P_k = P_{new}$
13: **return** $\mathcal{C}_{MOO}$

class, called outliers, which are projected far away from those kernels, can be considered as negatives. The result of this kind of classifiers is a not so high ratio of CD, but a very low ratio of FA. Curve ROC2 in fig. 2(b) designs this kind of behavior. When P is heterogeneous, Adaboost has a hard work grouping the positive samples on the classification space, generating largest boundaries. This results on a highest ratio of CD and, at the same time, an increase of the FA. Curve ROC1 in fig. 2(b) illustrate this behavior compared to ROC2.

Algorithm 1 trains a pool of $\mathcal{N}$ Cascades of Boosted Classifiers: $\mathcal{C}_{MOO} = \{\mathbf{C}_1, ..., \mathbf{C}_\mathcal{N}\}$. Function $TrainCascade()$ on line 4 follows the guidelines of [12] to train all $\mathbf{C}_i$. It uses as input argument positive datasets P_k with growing heterogeneity as i goes from 1 to $\mathcal{N}$. Therefore, $\mathbf{C}_\mathcal{N}$ will have wider boundaries on the classification space than $\mathbf{C}_1$. It can be done by removing from P_k the positive samples placed at the center of the kernels on the classification space by $\mathbf{C}_k$. Functions $ComputeOutputScores()$ and $SortIndexIncreassingOrder()$ obtain and sort the scores of all the samples in P_k using $\mathbf{C}_k$, placing the highest scores at the end of the list $oidx$. Variable p, which represents the number of positive samples, is decreased on line 8 by a factor or d_{min}^n $(d_{min} < 1)$. Then, P_{new} dataset is created by the first p samples of list $oidx$, and will be the next positive dataset to train $\mathbf{C}_{k+1}$.

3 Experiments and Results

Training and Test Datasets. The positive dataset $\mathcal{P}$ employed on the training procedure is composed of rectangular images containing a person from video sequences captured at a street corner, as shown fig. 1(a). 6,726 positive samples were obtained by flipping the patches on the vertical axis.

The negative set used to train the classifiers is the PASCALVOC 2012 dataset composed of 7,166 images without persons. The INRIA person negative set (1,570 images) is also employed but for the construction of the ROC curves and the definition of the Pareto Front.

The pedestrian detection systems are tested on the GSDatasets, which consist of view of a street corner capturing pedestrians while crossing the street.

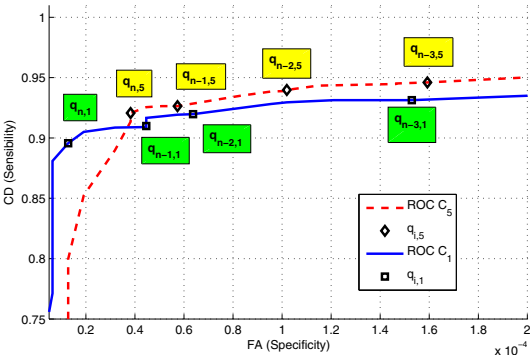


Fig. 3. The figure shows ROC curves from $\mathbf{C}_1$ and $\mathbf{C}_5$, and the operational points $q_{i,1}$ for cascade $\mathbf{C}_1$ and $q_{i,5}$ for cascade $\mathbf{C}_5$.

The datasets are public and available at http://pablonegri.free.fr/Downloads/GSdataset-PANKit.htm. They consist on two sequences of two minutes where each pedestrian on the sight has its bounding box and a unique label. This sets can be employed to test detection and tracking systems on outdoors sequences. In sequence GS06, there are five persons crossing the street, generating 1,157 position to detect for the classifier. Sequence GS54 has 3,644 positive positions generated by 16 persons which cross the street. The remaining pedestrians of both sequences are not evaluated by the detection system.

The training was performed using a 3 fold-cross validation technique. The total positive base $\mathcal{P}$ is divided randomly in three datasets $\{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$. Each training of $\mathcal{C}_{MOO}$ employs two of those sets to construct the positive dataset P. The remaining set, it can be denominated as V, is employed to compute the ROC curve in order to characterize the behavior of $\mathcal{C}_{MOO}$.

For each $\mathcal{C}_{MOO}$ were trained $\mathcal{N} = 5$ $\mathbf{C}_i$ following algorithm 1. This value of $\mathcal{N}$ was chosen based on the number of positive samples that were discarded at each iteration. It depends on variable d_{min} that is equals to 0.995 in our experiments. For $\mathcal{N} = 5$, the remaining number of positive samples is enough to train the last cascade $\mathbf{C}_5$: the number of positives training $\mathbf{C}_1$ is 4,484, and 3,253 for $\mathbf{C}_5$. The variable f_{max} has a value of 0.4.

A modification to the $TrainCascade()$ function of alg. 1 was introduced to train $\mathbf{C}$. The *regular* version uses the same positive dataset on the Adaboost algorithm for training and validating the strong classifier. This set changes at each stage of the Cascade training. On the modified version fix, the positive dataset is split on two sets for the training and the validation. The validation set is the same (is 'fixed') during all the training of $\mathbf{C}$.

The architecture of *regular* $\mathbf{C}_1$ has 12 stages C_i, while the *fixed* $\mathbf{C}_1^{fx}$ has 20 stages. The stage number of both versions of the Cascades increase within the training of the pool, because the heterogeneity of the positive datasets.

Figure 3 draws ROC curves obtained from a set $\mathcal{C}_{MOO}$ using dataset V (2,242 positive samples) and 100,000 negative patches from INRIA negative person set.

This figure shows the ROC of $\mathbf{C}_1$ and $\mathbf{C}_5$. Pareto Front can be estimated choosing the operational points of one ROC which locally dominates the operational points of the other ROC, maximizing the DC and minimizing FA.

Implementation of the MOO System. This section proposes a methodology to select the *Pareto Optimal Solution* depending on the dynamic of the scene.

From fig. 3 the Pareto Front will be composed of the non-dominanted operational points from both classifiers $\mathbf{C}_1$ and $\mathbf{C}_5$. When the dynamic of the scene changes, it is possible to chose another operational point of the same $\mathbf{C}_i$ or change for the other classifier.

To simplify the operation, two operational points of the Pareto Front are applied depending on the state of the traffic light. Because the objective is to increase the detection of pedestrians walking in front of the stopped vehicles the classifier should have wide boundaries on the classification space, and high CD rate. However, the continual use of this operational point on the Pareto Front, which also has high FA rate, will droops the performance of the detection system. When pedestrians stop crossing the street because the vehicles are circulating, the Pareto Solution can change for another operational point which belong to a classifier with narrow boundaries on the classification space. As the dynamic of the vehicles is governed by their traffic light, the change of the operational point on the Pareto Front will also be determined by their states:
- Green Traffic Light: operational point $q_{green} \rightarrow \mathbf{C}_1\{q_{n,1}\}$ to minimize FA.
- Red Traffic Light: operational point $q_{red} \rightarrow \mathbf{C}_5\{q_{n,5}\}$ or $\mathbf{C}_5\{q_{n-1,5}\}$, or $\mathbf{C}_5\{q_{n-2,5}\}$, to maximize DC.

The performance is evaluated using: *CD* as the number of pedestrians correctly detected, the *Miss Rate* as the percentage of non-detected pedestrians, *FA* the total amount of false alarms on the set, and the *Average Precision Ratio* (AP) obtained from the Precision-Recall curve at the choose operational point.

Table 1 presents the results of the MOO system compared to the regular Cascade of Boosted Classifiers $\mathbf{C}$, and the 'fixed' version $\mathbf{C}^{fx}$. The results depicted on table 1 of both versions of $TrainCascade()$ function, exhibit that the 'fixed' version has a better performance, and the implementation of the MOO system shows a better performance than the classic implementation.

Table 1. Detection results using different systems.

	GS06 (1,157 positives)				GS54 (3,644 positives)			
Detector	CD	Miss Rate (%)	FA	AP	CD	Miss Rate (%)	FA	AP
$\mathbf{C}$	790	31.7	**160**	67.8	2194	39.8	408	58.1
$\mathbf{C}^{fx}$	836	27.7	354	70.5	2468	32.2	749	64.9
$\mathcal{C}_{MOO}\{q_{n,1}, q_{n,5}\}$	805	30.4	238	67.7	2308	36.6	471	60.5
$\mathcal{C}_{MOO}\{q_{n,1}, q_{n-1,5}\}$	812	29.8	323	68.1	2345	35.6	611	60.9
$\mathcal{C}_{MOO}\{q_{n,1}, q_{n-2,5}\}$	817	29.3	474	68.0	2389	34.4	775	61.4
$\mathcal{C}^{fx}_{MOO}\{q_{n,1}, q_{n,5}\}$	852	**26.3**	460	**71.7**	2554	29.9	**842**	67.1
$\mathcal{C}^{fx}_{MOO}\{q_{n,1}, q_{n-1,5}\}$	854	26.1	523	71.6	2570	29.4	907	67.3
$\mathcal{C}^{fx}_{MOO}\{q_{n,1}, q_{n-2,5}\}$	854	26.1	562	71.5	2581	29.1	956	67.4

As expected, MOO systems maximize the number of CD, minimizing the Miss Rate, while the number of FA increase within acceptable values. For example, $\mathcal{C}_{MOO}^{fx}\{q_{n,1}, q_{n-1,5}\}$ system increases the number of CD by 18 samples on the GS06 dataset, and the FAs grows about 100, meaning one FA each 10 frames. The advantage of the MOO system is better appreciated on the GS54 dataset, while the FAs increase, again, by 100 samples, the system detect almost one hundred additional pedestrians, representing 2.5 % in comparison with the classic implementation. For detection systems, a greater number of CD is more significant as shown the highest values of AP. Thus, this combination can be choose as the best for this application.

4 Conclusions

This article proposes a Multi-Objective Optimization System applied to pedestrian detection on outdoor scenes complexes with cluttered backgrounds. A pool of classifiers is trained using different combination of positives datasets. Depending on the dynamic of the scene, different operational points corresponding to locally non-dominated solutions of the Pareto Front are applied to improve the system performance. The perspectives will be oriented to develop a methodology to optimize the choose of the positive samples to train the pool of classifiers.

Acknowledgments. This work was funded by the ACyT A14T24 (UADE), and the PICT-BICENTENARIO 2283 (FONCYT).

References

1. Bradley, A.: The use of the area under the roc curve in the evaluation of machine learning algorithms. PR **30**, 1145–1159 (1997)
2. Cabezas, I., Trujillo, M.: A method for reducing the cardinality of the pareto front. In: Alvarez, L., Mejail, M., Gomez, L., Jacobo, J. (eds.) CIARP 2012. LNCS, vol. 7441, pp. 829–836. Springer, Heidelberg (2012)
3. Chatelain, C., et al.: A multi-model selection framework for unknown and/or evolutive misclassification cost problems. PR **43**(3), 815–823 (2010)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. CVPR **1**, 886–893 (2005)
5. Felzenszwalb, P., Girshick, G., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI **32**(9), 1627–1645 (2010)
6. Li, W., Liu, L., Gong, W.: Multi-objective uniform design as a svm model selection tool for face recognition. Expert Systems with Applications **38**, 6689–6695 (2011)
7. Negri, P.: Estimating the queue length at street intersections by using a movement feature space approach. IET IP **8**(7), 406–416 (2014)
8. Negri, P., Goussies, N., Lotito, P.: Detecting pedestrians on a movement feature space. PR **47**(1), 56–71 (2014)

9. Negri, P., Lotito, P.: Pedestrian detection using a feature space based on colored level lines. In: Alvarez, L., Mejail, M., Gomez, L., Jacobo, J. (eds.) CIARP 2012. LNCS, vol. 7441, pp. 885–892. Springer, Heidelberg (2012)
10. Rosales-Pérez, A., Gonzalez, J.A., Coello-Coello, C.A., Reyes-Garcia, C.A., Escalante, H.J.: Evolutionary multi-objective approach for prototype generation and feature selection. In: Bayro-Corrochano, E., Hancock, E. (eds.) CIARP 2014. LNCS, vol. 8827, pp. 424–431. Springer, Heidelberg (2014)
11. Rosales-Pérez, A., et al.: Surrogate-assisted multi-objective model selection for support vector machines. Neurocomputing **150**, 163–172 (2015)
12. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. CVPR **1**, 511–518 (2001)