AUTOMATIC PROCESS TO BUILD A CONTEXTUALIZED DETECTOR

Thierry Chesnais[1], Nicolas Allezard[1], Yoann Dhome[1] and Thierry Chateau[2]
[1]*CEA, LIST, Vision and Content Engineering Laboratory, Point Courrier 94, F-91191 Gif-sur-Yvette, France*
[2]*Lasmea, UMR6602, CNRS, Blaise Pascal University, Clermont-Ferrand, France*
{thierry.chesnais, nicolas.allezard, yoann.dhome}@cea.fr, thierry.chateau@lasmea.univ-bpclermont.fr

Keywords:
video surveillance, object detection, pedestrian detection, semi-supervised learning, oracle.

Abstract:
This article tackles the real-time pedestrian detection problem using a stationary uncalibrated camera. More precisely we try to specialize a classifier by taking into account the context of the scene. To achieve this goal, we introduce an offline semi-supervised approach which uses an oracle. This latter must automatically label a video, in order to obtain contextualized training data. The proposed oracle is composed of several detectors. Each of them is trained on a different signal: appearance, background subtraction and optical flow signals. Then we merge their responses and keep the more confident detections. A specialized detector is then built on the resulting dataset. Designed for improving camera network installation procedure, the presented method is completely automatic and does not need any knowledge about the scene.

1 INTRODUCTION

In computer vision, the problem of real-time (at least 10 frames by second) and robust object detection, in particular pedestrian detection, is still a hot research topic. These algorithms are useful in video surveillance context or in Advanced Driver Assistance Systems. Some of the last advances made in this field have been publied in (Enzweiler and Gavrila, 2009) (Gerónimo et al., 2010) (Dollár et al., 2011). The variability of appearance in the pedestrian class is important (size, posture, lighting). So it is important to consider the context of the scene to build a specialized detector.

Classical approaches to detect objects are based on machine learning. Support vector machine (Vapnik, 1995)(Dalal and Triggs, 2005) and boosting algorithms are the mainly used methods. These processes consist to extract the best discriminative features between pedestrian and background from a labeled training dataset. Then the obtained detector compares the selected features of a new image with these of the database to predict the presence of a pedestrian.

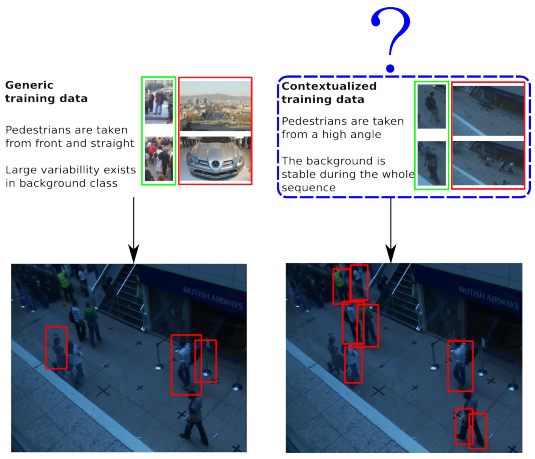


Figure 1: A detector trained on a generic dataset (left) reach lower performances than a contextualized classifier (right) when the point of view of the learning and the detection bases are too different.

However to reach the best detection performances with these methods, the training data must be as rich as possible and own features

similar to those computed during the detection step. Consequently it is essential for the training dataset to be contextualized. Building such a base with thousands examples well annotated and aligned is an expensive manual task. That is why it is not realistic to collect examples for each camera during the deployment of a CCTV equipment. In this case it is often necessary to use a classifier train on a generic training dataset, hoping this process will not degrade performances too much (see the figure 1).

During the last years several approaches have been proposed to tackle the problem of automatically building a training dataset in order to exploit large amount of images recorded by cameras. Semi-supervised methods are often part of the proposed solutions in bibliography, because they are designed to use, directly in the training set, labeled but especially unlabeled data.

The training dataset is called contextualized when it contains a lot of specifics information coming from the scene. The data could be integrated in the specialized classifier in several ways:

- collecting a large database to train a one shot classifier is the principle of **offline methods**;
- training the classifier as soon as new samples are available, is the principle of **online methods**. These latter have been generalized in computer vision by (Grabner and Bischof, 2006).

Our goal is to propose a new semi-supervised method. Using an oracle will permit to automatically build a classifier which will be adapted to the particular context of the scene. We choose to train our detector with an offline method for two reasons. Firstly our procedure occurs at the time of a camera network installation. Although we have all necessary time to obtain and treat a lot of examples, we prefer to avoid training an online classifier during exploitation and keeps all computer resources for detections. Secondly even if there are some online strong methods (Leistner et al., 2009), there is still a risk of drifting that seems not compatible with a long-term use.

In this study, we focus on how to build an oracle. After having detailed the most used semi-supervised methods, we describe, in the third part our strategy to create the oracle. The part 4 presents an evaluation of the proposed process consisting in an analysis of the behaviour of the oracle and a comparison with a state of art classifier.

2 STATE OF ART

There are a lot of families of semi-supervised methods. The most common approaches are the self-learning ones, the co-training ones and the methods based on an oracle.

The **self-learning** (Rosenberg et al., 2005) approach consists in using the output of a classifier to annotate a new example. If a classifier is very confident about a sample, this latter is added to the base. This method lacks of robustness suffering from a drift problem. Mislabeled examples will indeed disrupt the classifier, change its behaviour for the next samples and in consequence make the phenomenon worse. Moreover if the confident threshold used to separate classes is too low, a lot of false positives will be incorporated in the base. On the contrary if the threshold is too high, only perfectly identified samples, the ones containing little information, are kept.

The **co-training** introduced by (Blum and Mitchell, 1998) is a formalism in which two classifiers are trained in parallel. Each of them uses a different and independent part of the data. For example (Levin et al., 2003) train two classifiers, one on appearance signal and the other one on background subtraction signal. The co-training algorithm uses the fact that an example must have the same label with both classifiers even if they are not trained on the same data. If one of the detectors labels with confidence a sample, the other one being unsure, the sample is incorporated in the base of the second classifier. During the training phase, each classifier improves its performance thanks to the confidence of the other one. Endly we obtain two well trained detectors. Even if detectors are independent, the problem here is, like with the self-learning, the outputs of the classifiers are still directly used to label samples. Drift problem are not completely excluded because parts of the data are seldom independent.

Methods based on an oracle use an external entity to build a dataset. This entity annotates all examples before adding them in the training data. Final detector does not affect the outputs of the oracle reducing the drift problem. The capacity of an oracle to find good samples without error determines the performance of the final classifier. If the oracle does not work well on a video the whole system is useless. A lot of different classifiers have already been proposed.

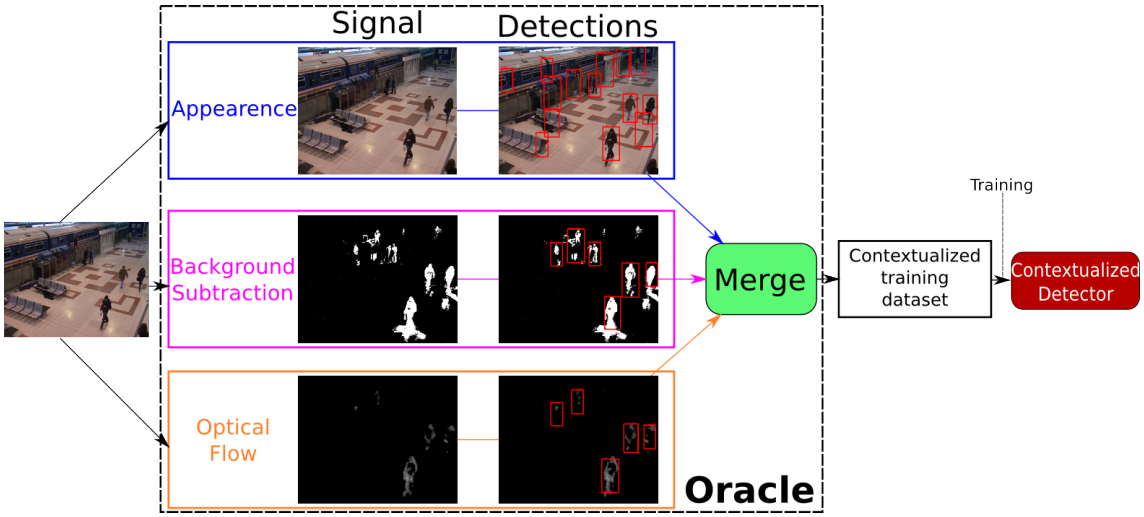


Figure 2: Oracle diagram. The oracle is formed by three independent classifiers based on appearance, background subtraction and optical flow signals. Each one gives a set of detections which will be merged (see figure 3) to build a contextualized training dataset. The resulting output is used to create a contextualized powerful detector.

(Wu, 2008) uses a part based classifier applied on appearance signal. If the oracle find some pedestrian parts, the sample is added in the training data. One drawback of the method is the fact that the oracle is composed of only one classifier dealing with only one signal. Another problem is the difficulty of detecting pedestrian parts and merging them. To add robustness, (Stalder et al., 2009) uses an oracle with several stages. First step consists in detecting people in the picture. In a second part trackers are initialized on this detection. The author's goal is to obtain some spatio-temporal continuity between oracle detections to incorporate samples which have not been detected. Contrary to Wu's approach, this allows to find some hard examples. A last stage uses 3D information. The main drawback of this scheme is its structure. If a stage failed, errors are inevitably passed to the next one without any possibility to correct them.

We propose an oracle working in a no-sequential way in order to improve robustness.

3 CONTEXTUALIZATION OF A DETECTOR

In this part we describe the different steps of our method to build an oracle. In the same way than co-training approach, this latter is formed by several classifiers working on different and independent signals. Unlike Stalder's approach (Stalder et al., 2009) which uses a sequential oracle, our method has the ability, after a merging phase, to suppress bad detections given by each of the signals. This capacity improves the training set.

3.1 Oracle

3.1.1 Specifications

The oracle must automatically annotate a video which means finding relevant observations and the associated labels.

The oracle is a pedestrian detector with characteristics different from the contextualized detector. In addition to be real-time, this latter must detect as many pedestrians as possible with minimal false positive rate. In other words, it must have both a high recall: $\frac{\text{number of good detections}}{\text{number of pedestrians}}$ and a high precision: $\frac{\text{number of good detections}}{\text{number of detections}}$. For the oracle, it is possible to release some constraints. It does not need to be real-time since our method counts two steps. The first one could last for a long time. Moreover our purpose is to build a training dataset. It is not penalizing to miss some pedestrians since the video is long enough to offer a lot of positive examples. To sum up, the oracle could have a lower recall than the final detector, but in order to minimize the label noise in the contextualized base, it must be as precise as possible.

3.1.2 Constitution

To satisfy these specifications we decide to use a combination of elementary classifiers (figure 2). Each of them is trained on a different and independent signal like in the co-training method. Therefore a merging phase (figure 3) is able to correct some errors, by cross-validating responses given by each classifier.

In this article we have exploited three signals: appearance (descriptor based on gradient), background subtraction (Stauffer and Grimson, 1999) and optical flow (Black, 1996). Each classifier is based on a different descriptor implying that it uses a different generic training data. The three classifiers are running in parallel.

3.1.3 Building a Contextualized Training Data

The three previously trained classifiers build a base by scanning context images. For every position and scale in an image, each classifier gives a detection score. The next step of the process implies to merge the confidence maps. Unfortunately the classifiers are *a priori* independent and their outputs are not comparable. There are two solutions: working directly with the confidence maps after normalizing them to be sure they are comparable; or working on the detections given by each classifier after a clustering. We choose the last option. In a similar manner as (Dalal and Triggs, 2005), we use a meanshift to group all the boxes which have a positive score. Each resulting fused detection has a score which corresponds to the sum of the group boxes' scores. For an image we get a set of detections (box and score) for each classifier.

The **positive examples** correspond to observations with which the generic classifiers are confident. The merging phase between the detectors is a delicate step. If this process is too restrictive, some hard and thus interesting examples can be missed. On the contrary if this step is too weak, the training data would be polluted by lots of false positive samples.

A detection is incorporated in the base only if it appears in the output of several classifiers. A majority vote is done as explained in figure 3. First a greedy association is performed between the detections coming from the appearance detector and these coming from the background subtraction one. Only the associated detections coming from appearance are added in the contextualized base. The association is performed using an overlap criterion between detections. Like (Everingham et al.,) we use the criterion: $\text{sim}_{\text{detection}}(B_1, B_2) = \frac{Area(B_1 \cap B_2)}{Area(B_1 \cup B_2)}$. If two boxes have a similarity under 0.5, they are considered not-linked. A second association is realized between the remaining detections coming from appearance and these coming from the optical flow detector. As previously, the associated detections coming from the appearance are integrated to the base whereas the not associated boxes are thrown. In fact this vote corresponds to a check on detections from appearance signal with these from background subtraction and optical flow used as validation. These last two detectors are characterized by their lack of precision whereas the one based on appearance signal gives a more accurate response. That is why the merge step favours the detections coming from appearance and validate them with the two other classifiers.

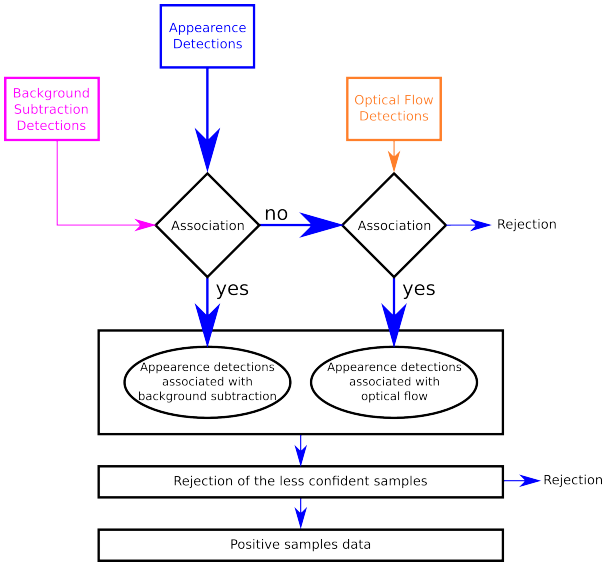


Figure 3: Workflow, illustrating the three classifiers merge process, designed to provide positive samples.

A last filter on the boxes score is done in the training dataset. As all detections come from appearance signal, scores are comparable. During this final step about 50% of the less confident examples are suppressed.

After selecting the positive examples, we need to compute the **negative samples**. Our strategy

consists in choosing random boxes in the whole image except in areas where there is at least a positive detection. As the oracle has a low recall, it does not detect all pedestrians. Some can be incorporated in the negative base. It is rather unlikely because there are a lot more negative examples in an image than positive ones.

As we are working on a static scene, a lot of observations are similar. The risk here is to create a base not rich enough with to few hard examples. To deal with this problem, we add in the base some examples which contains pedestrian parts. They correspond to an image example intersecting with a oracle detection. However both samples must not overlap too much and verifying the previously defined criterion:
$\text{sim}_{\text{detection}}(B_{\text{pedestrian}}, B_{\text{negative example}}) < 0.5$

3.2 Building a Contextualized Detector

To create a contextualized detector, we need to train a new classifier with the contextualized training data.

A possibility is to do an offline training using a dataset containing the three signals and let the boosting algorithm choose a good combination between them. With this approach, a maximum of information from the training set is exploited. In our experiments we remark than the final detector obtained with this method is not significantly performing better than a detector only trained on an appearance signal, however the computing time increases a lot from one to the other solution. In consequence we choose a simpler classifier based only on the appearance signal as final detector.

4 EVALUATIONS

In this section we do a method assessment split in two parts. Firstly we study characteristics of the oracle presented in section 3. Secondly we compare performances of the final detector with a state-of-art one and show its competitiveness.

The algorithm has been tested on the freely available datasets : PETS 2006[1], PETS 2007[2].

We evaluate our system with the method described in (Agarwal et al., 2004) and illustrated with precision-recall curves. Precision is $Pr = \frac{TP}{TP+FP}$ and recall is $R = \frac{TP}{P}$ where TP is the number of true positives, FP the number of false positives and P the number of pedestrian. We plot the curves R depending on $(1 - Pr)$. The optimal point is located in (0, 1). F-Measure is defined as $FM = 2.\frac{Pr.R}{Pr+R}$.

The similarity criterion used between the ground truth (GT) and a test box (B) before the clustering is:

$$\text{sim(GT, B)} = \frac{(GT_{cx} - B_{cx})^2}{(0.5 \times w(GT))^2} + \frac{(GT_{cy} - B_{cy})^2}{(0.5 \times h(GT))^2}$$

with:

- cx and cy corresponding respectively to the abscissa and to the ordinate of the centre of a box,
- $w(GT)$ and $h(GT)$, respectively the width and the height of the ground truth box.

Two boxes are similar if sim(GT, B) ≤ 1. This criterion could be seen as a definition of an ellipse around the center of a ground truth box. If the center of a detection fall in this ellipse it is considered as positive. If two detections are linked to the same ground truth box, only one true positive sample is counted. Others boxes correspond to false detections.

4.1 Characteristics of the Oracle

In this paragraph, we study the oracle characteristics by checking if it corresponds to the specifications. As previously explain we train three generic classifiers on appearance, background subtraction and optical flow signals. Each classifier is trained with 400 rounds of boosting (Real-AdaBoost using decision stumps (Friedman et al., 1998) (Schapire and Singer, 1999)) without cascade. With the same detection threshold, not using a cascade increases the recall of the detector (more detections) but decreases its precision (more false positives). This latter is optimized by the classifiers merging phase.

The appearance classifier uses a descriptor based on gradient. We use the same descriptor for the optical flow. Horizontal and vertical components of optical flow correspond to horizontal and vertical components of the gradient of appearance. However for the background subtraction descriptor we decide to use Haar wavelets. Unfortunately with these features, it is impossible to know if a homogeneous area is a part of an

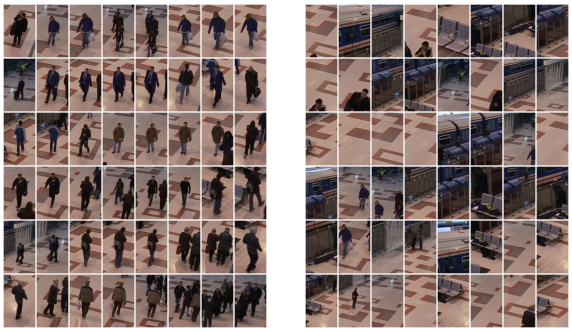
[1] http://www.cvg.rdg.ac.uk/PETS2006/
[2] http://www.cvg.rdg.ac.uk/PETS2007/

object or just background. To solve this problem we add in our descriptor the mean of the current wavelet window.

We collect, from the INRIA person dataset[3], 2417 positive and 25742 negative examples to train our appearance detector. This dataset have no temporal information. Consequently we build two new independent datasets for the others classifiers. We train the background detector with 787 positive and 6466 negative samples and the optical flow detector with 776 positive and 8000 negative examples. Detection threshold are set to 0 for all classifiers.

4.1.1 PETS 2006

We evaluate our method on the view 4 of the PETS 2006 dataset. The examples of the training data are coming from S2-T3-C and we test the final detector on about 1000 frames from S7-T6-B.



(a) positive examples (b) negative examples possibly containing pedestrian parts

Figure 4: Contextualized training data extracted from the PETS 2006 dataset.

The figure 4 shows some samples of the training data after the merge of the classifiers. For positive examples a large majority of thumbnails are effectively a pedestrian. However there are still two main issues:

- When several pedestrians are close, the clustering could not always separate them correctly and that tends to misaligned the resulting example,
- The size of the thumbnail is not always adapted to the object.

[3]http://pascal.inrialpes.fr/data/human/

As we hoped, almost all the negative examples correspond to areas without pedestrian or include limited pedestrian parts.

On the training video (S2-T3-C - 4), the oracle obtain a recall of 0.16 and a precision of 0.99. Notice that because of the filtering step after the merging phase, the recall value depends on the number of frames we use to build the training dataset. As expected the oracle has a very high precision. The result are obtained without any knowledge of the scene (like the ground plane or a 3D model of the scene) and without any threshold since all detectors have the same detection threshold (the threshold is fixed to 0).

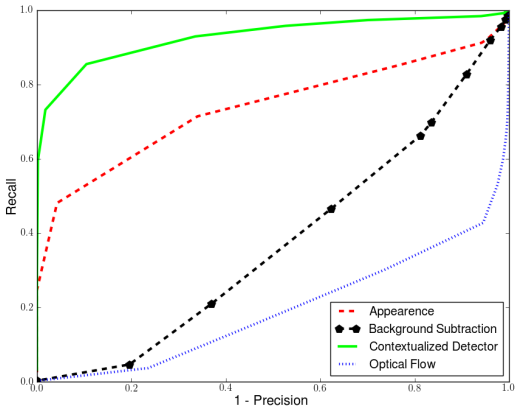


Figure 5: Precision-Recall curves illustrating detection performances on PETS 2006 for the three classifiers forming the oracle and the contextualized detector.

We can see on the figure 5 the precision-recall curves of each detector involved in the oracle and the curve of the contextualized detector. This latter only uses appearance information and is trained in the same way than the oracle appearance detector. The training data is the only difference between them. 1800 positive and 8000 negative examples are kept after the filter step in the classifier merging process and are used for the training.

The table 1 gives recall and precision values for each classifier where its f-measure is maximized. The precision of the background subtraction and optical flow classifiers are weak. This can be explained by the fact that they are not very discriminant. Their detections are spread around a target and often, two close pedestrians are confused after clustering. As we explain, when we

Table 1: Results of the different classifiers applied on PETS 2006 - S7-T6-B - 4

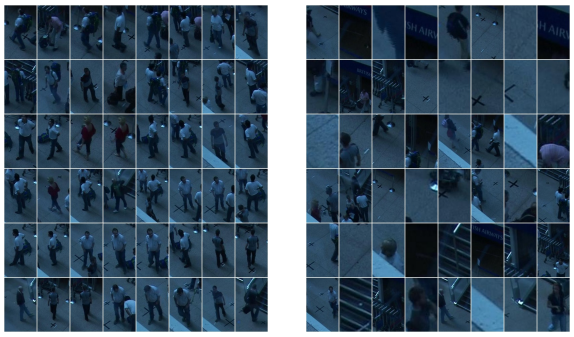
	Recall	Precision	F-Measure
Appearance	0.71	0.66	0.69
Background	0.47	0.38	0.42
Optical flow	0.30	0.26	0.28
Oracle	0.49	**0.99**	0.65
Contextualized detector	0.85	0.90	**0.87**

have presented the merging step, background subtraction and optical flow detectors could be seen as presence captors, reliable to predict pedestrian presence but inaccurate in location, whereas the appearance detector is less robust but its detections are well localized.

4.1.2 PETS 2007

We test our algorithm on the third view of PETS 2007. It is to notice that the pedestrians are shot with a high angle and are generally leaned in this video. This point of view is interesting to illustrate the interest of our approach because of their differences with the INRIA dataset (pedestrians are taken from the front and are straight), used to train our initial classifier based on appearance.

The contextualized training data has been built on the sequence called S03. Each detector is evaluated on the 1000 first images of the fifth video.



(a) positive examples (b) negative examples possibly containing pedestrian parts

Figure 6: Contextualized training data extracted from the PETS 2007 dataset.

The figure 6 shows samples from the training data collected after the fusion of classifiers forming the oracle.

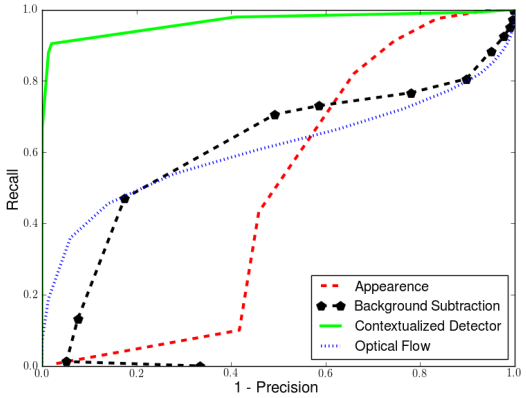


Figure 7: Precision-Recall curves on PETS 2007 for three classifiers in the oracle and the final detector.

The figure 7 presents the recall-precision curves for each classifier of the oracle and the contextualized detector.

Table 2: Results of the different classifiers applied on PETS 2007 - S05 - 3

	Recall	Precision	F-Measure
Appearance	0.82	0.34	0.48
Background	0.47	0.82	0.60
Optical Flow	0.54	0.72	0.62
Oracle	0.40	**0.99**	0.57
Contextualized detector	0.90	0.98	**0.94**

In the same way than PETS 2006, the table 2 contains precision and recall values for each classifier where its F-measure is maximized. In the oracle case, results are only given on positive examples.

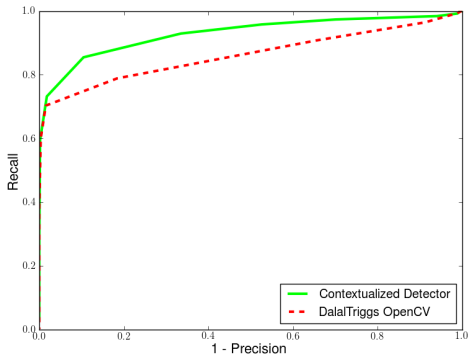
Contrary to previous video, classifiers based on background subtraction and optical flow signals have better performances than the one on appearance signal. It could be easily explained:

- Examples collected on this sequence and these coming from the training data have very different appearances. Consequently a classifier using only this signal has poor performances.
- There are several groups of people in this video. Classifiers which are not very discriminant are not too penalized because even if a detection is far from a pedestrian, an other one could be caught in the detection window.

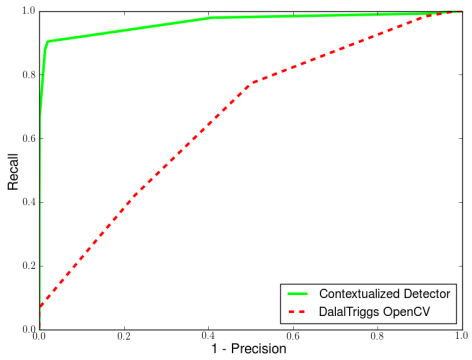
As curves prove it on these two datasets, a contextualized detector reaches better recall and precision than the generic oracle only based on appearance.

4.2 Performances of the Contextualized Detector

In this part we compare performances of the contextualized detector with a state of art one. We choose the Dalal and Triggs (Dalal and Triggs, 2005) detector available in OpenCV. We apply the same evaluation criterion.



(a) PETS 2006



(b) PETS 2007

Figure 8: Recall-precision curves of our final detector (green) and the one of Dalal and Triggs (red) on PETS 2006 and 2007.

Curves on the figure 8 show results on the two sequences used in this paper. On the PETS 2006 video both classifiers have similar results. Dalal and Triggs detector already reaches a high level of performances. Therefore although our approach improves the detection rate, both classifiers could be use on this sequence. On the contrary when the video and the learning dataset have a very different point of view, Dalal and Triggs detector is not very successful in detecting most of pedestrians. In this case, our method, using an oracle formed by basic classifiers, can achieve good performances because the detector is contextualized.

When the training data and the scene are too different, a contextualized detector improves results significantly.

5 CONCLUSIONS

We proposed a semi-supervised method. It is aimed at automatically training a contextualized detector. To achieve this goal we create an oracle composed of several classifiers. Each of them works on a distinct signal. A merging step of the different responses is then done to build a specialized training database. This set is then used to train a final detector incorporating contextualized information.

Even if our approach gives some good results, several improvements are possible.

- As previously notice, classifiers based on background subtraction and optical flow are not very precise. They are less discriminant in fact and they tend to merge proximate detections. To mitigate this phenomenon, it is possible to use calibrated cameras in order to remove aberrant detections. Unfortunately this requires a manual step during the camera network installation. That is why we do not use it.
- In this study, we choose to build an oracle with three signals. However it is possible to use other ones. For example if we have a stereo camera, it is possible to learn a classifier directly on the disparity maps and add it in the oracle.
- We choose to use an offline algorithm to train our final detector. However it could be interesting to study the behavior of our system with an online training. This has the advantage to allowed a regular update of the classifier, in the hope to tackle the problem of changes in the scene (lighting, background...).

REFERENCES

Agarwal, S., Awan, A., and Roth, D. (2004). Learning to detect objects in images via a sparse, part-based representation. *Pattern Analysis and Machine Intelligence*.

Black, M. (1996). The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields. *Computer Vision and Image Understanding*.

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Int. Conf. on Computer Vision and Pattern Recognition*.

Dollár, P., Wojek, C., Schiele, B., and Perona, P. (2011). Pedestrian detection: An evaluation of the state of the art. *Pattern Analysis and Machine Intelligence*.

Enzweiler, M. and Gavrila, D. M. (2009). Monocular pedestrian detection: Survey and experiments. *Pattern Analysis and Machine Intelligence*.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results.

Friedman, J., Hastie, T., and Tibshirani, R. (1998). Additive logistic regression: a statistical view of boosting. *Annals of Statistics*.

Gerónimo, D., López, A. M., Sappa, A. D., and Graf, T. (2010). Survey of pedestrian detection for advanced driver assistance systems. *Pattern Analysis and Machine Intelligence*.

Grabner, H. and Bischof, H. (2006). On-line boosting and vision. In *Int. Conf. on Computer Vision and Pattern Recognition*.

Leistner, C., Saffari, A., Roth, P. M., and H., B. (2009). On robustness of on-line boosting - a competitive study. In *Int. Conf. on Computer Vision - Workshop on On-line Learning for Computer Vision*.

Levin, A., Viola, P., and Freund, Y. (2003). Unsupervised improvement of visual detectors using co-training. *Int. Conf. on Computer Vision*.

Rosenberg, C., Hebert, M., and Schneiderman, H. (2005). Semi-supervised self-training of object detection models. *IEEE Workshop on Applications of Computer Vision*.

Schapire, R. E. and Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. *Machine Learning*.

Stalder, S., Grabner, H., and Gool, L. V. (2009). Exploring context to learn scene specific object detectors. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*.

Stauffer, C. and Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In *Int. Conf. on Computer Vision and Pattern Recognition*.

Vapnik, V. N. (1995). *The nature of statistical learning theory*.

Wu, B. (2008). Part based object detection, segmentation, and tracking by boosting simple feature based weak classifiers.