MVA2009 IAPR Conference on Machine Vision Applications, May 20-22, 2009, Yokohama, JAPAN

# Face Blurring for Privacy in Street-level Geoviewers Combining Face, Body and Skin Detectors

Alexandre Devaux[1], Nicolas Paparoditis[1], Frédéric Precioso[2], and Bertrand Cannelle[1]
[1] Institut Géographique National - Laboratoire MATIS - Saint-Mandé France
[2] ETIS, CNRS, ENSEA - 95000 Cergy-Pontoise France
[1]firstname.lastname@ign.fr, [2]precioso@ensea.fr

## Abstract

*In the last two years, web-based applications using street-level images have been developing fast. In that context, privacy preservation is an unavoidable issue. We present in this paper a multi-boosting based approach to detect pedestrians in high resolution panoramics in order to blur their faces. This task is quite complex since these features vary in size, shape, color, and often are partially occluded, sometimes behind windows or inside cars, etc. Our strategy is thus based on the combination of two existing boosting algorithms detecting faces [1] and bodies [2] with a skin tone detection algorithm we developed. The results are quite encouraging for such an unconstrained data: 86.2% of true positives and an average of 2 false positive detections per image (2.1 MPixels). This combination solution provides much more robust results than each detection algorithm performed independently.*

## 1 Introduction

In the last two years, multimedia web-based applications using street-level ultra-high resolution images acquired by mobile mapping have been developing fast. Detecting pedestrians from these images is a killer issue especially for web-based geoviewers. Passers-by need to be detected and blurred out for legal privacy issues. Street level images are useful to enrich 3D city models generated from maps and/or aerial and satellite imagery, for model-based geoviewers. If several companies, like Blue Dasher Technologies Inc., EveryScape Inc., Earthmine Inc., Google$^{TM}$, try to provide their own multimedia solutions, Google$^{TM}$ is the only one that populated this new webservice wordlwide. As far as we know, Google$^{TM}$ is the only one which proposed a solution to take care of preserving privacy. Unfortunately, up to now, no information was provided on their pedestrian detection and face blurring system.

Our context is pretty similar to Google$^{TM}$'s one. We deal with huge panoramics (10176x5088 px; Figure 1) acquired by a mobile mapping system on large cities. People can be anywhere in the picture, with varying numbers, sizes, aspects (45°, frontal, profile), with varying light conditions (direct, diffuse, shadows), with often very strong occlusions due to trees, sign posts, cars, etc. We are thus in front of a very challenging problem.

The literature on pedestrian detection is rich. Nevertheless, most of the related work on pedestrian detection focus on real-time algorithms most of the time dedicated to obstacle detection and avoidance and often on small resolution (320 x 240 px) images. In the pedestrian detection domain, Dalal and Triggs [4] presented an efficient human detector using Histograms of Oriented Gradients and an SVM in 2005. This approach was rapidly optimized by Ivan Laptev [2] and Sabzmeydani et al. [5], substituing AdaBoost classifier to the original SVM. Nishida et al. [9] mixed Soft-Margin SVM which automatically select the best local-feature with Adaboost. Contributions on models have been proposed, Seemann et al [6] presented a generative object model which is scalable from general object-class detection to specific object-instance detection.

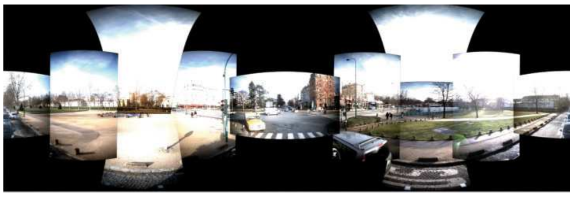

Figure 1: Example of a panoramic montage made with 10 cameras before post-processing

Most of pedestrian detectors are designed to detect only pedestrians. Indeed, most of the time they are not ideally designed to detect people on bikes, people sitting on a bench or in a car, or lying on the floor, etc, and most often they do not detect them. To deal with these free postures, an addition of a face and profile detector is necessary to increase the completeness of detection. The most famous face detector was presented in 2001 by Viola & Jones [1]. It was the first detector working in real-time with an excellent accuracy thanks to Haar features and AdaBoost. Many improved versions were proposed focusing on alternatives to Haar features and AdaBoost. In 2008, Yan et al [7] used LAB features with a feature-centric cascade algorithm which gave better results and increased the detection speed.

In the following, we first present our mobile mapping imaging system, then we present our detection strategy and the different existing boosting algorithms involved in the detection process and then we detail our skin tone algorithm. The last part presents the evaluation of the system showing some encouraging results, a detection rate of 86.2%.

## 2 Design of the mobile mapping system

The panoramic imagery we deal with is collected by a mobile mapping system which is composed of a set of ten full HD cameras mounted on a rigid frame.

The cameras are perfectly synchronized, mounted very closely, and have the same exposure times in order to build seamless panoramics. They have been chosen to have a high radiometric dynamic and a high signal to noise ratio (200-300) in order to manage the variations in illumination between the shadowed and the lightened sides of the street. The cameras are triggered in a way to acquire images at regular distance intervals (one panoramic per 3 meters). The images are georeferenced in a global reference frame with the help of an Inertial Navigation Systems (integrating 2 GPS, an Inertial Measurement Unit and an odometer) providing overall a submetric absolute localization. The intrinsic parameters of all the cameras were photogrammetrically estimated and the relative pose of the cameras are estimated by dense image matching on the image overlaps Craciun et al. [8] and automatic bundle adjustment. For each camera, a flatfield estimation and a color calibration using Greta targets is performed in order to retrieve realistic colors helpful for color based classifications.

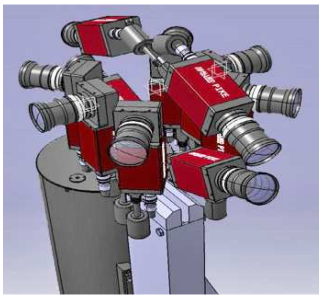

Figure 2: Camera system

## 3 Detecting faces and bodies

The environment in which we worked was the streets of the 12th district of Paris. An urban area where it is impossible not to photograph many people in the panoramic shots. To detect those people, we chose to combine face detection with pedestrian detection using Viola & Jones [1] algorithm, implemented in OpenCV library, and the algorithm of Laptev [2]. Then we added a skin detection algorithm we created in order to eliminate false positives. The system description is showed in Figure 3.

### 3.1 Appearance-based detectors

We used with the Viola & Jones [1] algorithm a face classifier and a profile classifier. The algorithm works on simple intensity variations with Haar Features, Laptev [2] algorithm works on gradients direction and the skin tone algorithm works on color intensity. The final result is the combination of four detectors working on different aspects of the data:

$$(HaarFace \cup HaarProfile \cup Laptev) \cap SkinTone^1 \Rightarrow H_2LS^2$$

The face detector and the pedestrian detector both use the Adaboost method to create a powerful cascade of classifiers. It is so fast that the face detector can be executed on any webcam in real time with a detection rate of nearly 100%. But if it takes less than 0.04 seconds on a 320x240 px image, on a 10176x5088 px image it takes obviously much more time. And if we parameter the algorithm to strengthen the detection, it lasts more than 3.5 minutes. Moreover, we have to do the same with the profile classifier. Thus, the algorithm lasts 7 minutes for just one panoramic image.

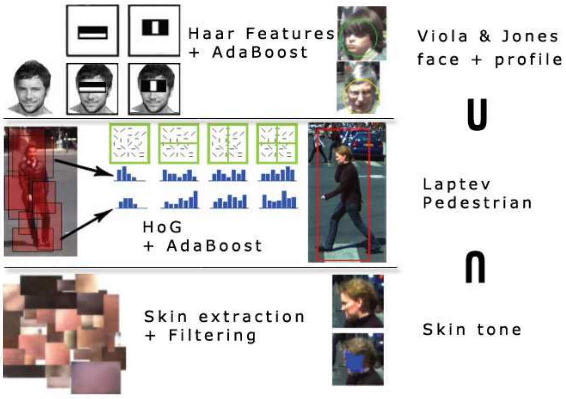

Figure 3: System description

Ivan Laptev [2] algorithm is very efficient, he showed his power on the PASCAL VOC challenge 2007 and it gives the best results for our system. It detects pedestrians using Histogram of Oriented Gradients (HoG) as descriptors and Adaboost for the intelligent learning (Laptev was inspired by the work of Kobi Levi and Yair Weiss [3]). The HoG are invariant to illumination and scale and can capture some geometric property very hard to get with linear descriptors like Haar. Figure 4 presents the results of the different algorithms on a crop of a single camera image.

### 3.2 Skin detection

Skin tone is often used for its invariance to orientation and size, gives an extra dimension compared to gray scale methods, and is fast to process. However it is also dependent on the illumination color, the ethnic group of the person, and many everyday-life objects are skin color like, i.e. skin color is not unique. We chose to use the skin tone because of its complementarity with the two other algorithms, and, because it is working on a completely different feature, the intersection with the results of the appearance-based algorithms should be more powerful.

The cameras we use are color-calibrated, so color (specially skin color) is rather stable. But as illumination varies, all objects color also varies. So illumination variations induce skin color variations, which increase the false detection rate.

Different tests were done. First, we tried a parametric method using intervals on Hue and Saturation values for the classifier. Results were encouraging, 92% of people skin detected but 50 000 false positive pixels per image (2.1MPx). Then, we developed an algorithm more efficient. We select from our images samples of skin from different ethnic groups with different illuminations (176 samples 12x12 px) and insert their RGB values in a set $X$. Then every time in the panoramic picture we encounter a pixel value existing in our set $X$ that means it should be some skin. Following this simple assumption we have already a high detection

[1] Haar Face/Profile is the algorithm of [1] with the face classifier and profile classifier. SkinTone is the skin detector.

[2] $H_2LS$: result of the four detectors combination. Two Haar-based detectors, Laptev's detector and the skin detector.

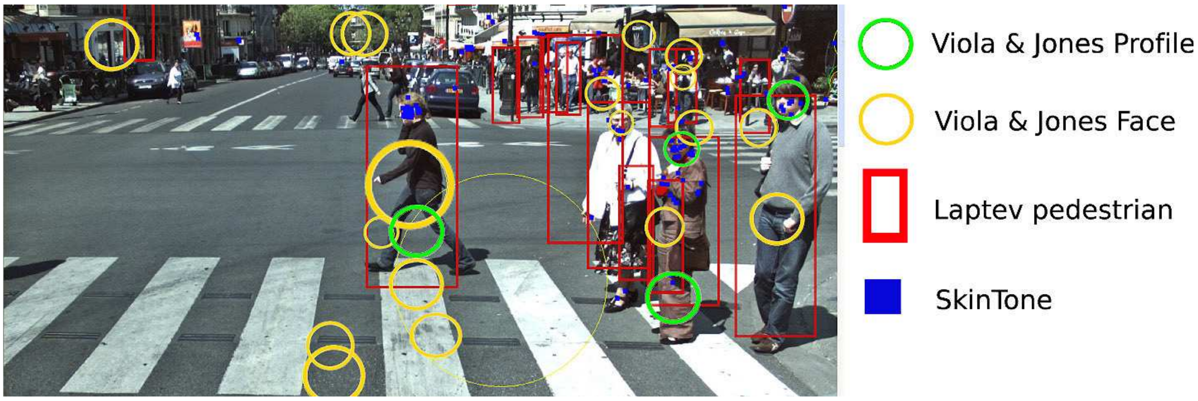

Figure 4: Results of the four detections on a part of a single camera photo

rate, in fact we have nearly 100% of detection. But we have also many false positives. The second aim is to decrease a lot the false detections filtering the skin set $X$.

With a learning set created of 75 photos, we filter our skin set $X$: We compute for all skin values $s$ of our set $X$ its frequency of apparition $f$ in the panoramics. Then we sort all skin values $s$ by $f$ in a vector $v$, beginning by the biggest frequency. For every value $s$ of $v$, we count the faces detected. If the number is inferior to the maximum detection rate we store this value in $X$ else we delete it (cf. algorithm 1). We consider, for the filtering step, that a face is detected if the referenced face contains at least $n$ skin pixels ($n = 3$ allows a strict filtering).

```
input  : RGB skin values from samples
output: A partition of the RGB skin values
$X \leftarrow$ RGB skin values from samples
$X \leftarrow Order_{byfrequency}(X)$
$R \leftarrow ComputeDetectionRate$
foreach $v$ in $X$ do
    $X \leftarrow X \backslash v$
    $R_2 \leftarrow ComputeDetectionRate$
    if $R_2 \neq R$ then
        $X \leftarrow X + v$
    end
end
```

**Algorithm 1**: Skin tone algorithm

This filtering reduces the number of Skin RGB Values from 20071 to 221. It surprisingly shows that detecting nearly all different faces is possible using only 221 values. A detection is validated if there is at least one skin pixel in a detected face or in the upper part of a pedestrian detection.

Another technic we added is a filter on lines. Some false skin detections often appear on pixel overlapping rectilinear edges (on the corner of wall stones, windows, etc.). This is due to the fact that the relative position of the image grid and the edge of the object vary along the object edge thus generating by integration a set of intermediate colors (of size depending on the edge slope) in between the colors of the objects on each side of the contour.

We thus filter out the pixel lying close to image edges using a Hough transform on lines in a window (10x10 px) around every skin pixel detected. If we find a line which is 8 pixels long at least we assume that the pixel detected was not skin because a face does not have that kind of geometry. At least we cannot find many lines in a face and if a face is near a post for example, we will not take into account the detected skin part of the face aside the post but other detections will remain.

It is important to emphasize the fact that if we detect just one skin pixel on a face it is enough since it is just used as a constraint with the other detectors (Face and body).

## 4 Experimental results and discussion

On the learning set, we searched for the best set of parameters optimizing the detection rate without taking into account the calculation time and the false alarms (in a maximum limit of 40% of the photo surface detected). Then, we launched the algorithms on 1150 photos HD (115 panoramics) referenced and achieved a detection rate of 89.5%. We can see the ROC curve on Figure 5. Table 1 presents the results for the different detectors and the combination $H_2L$.

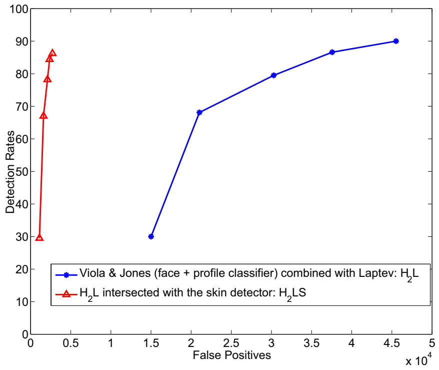

Figure 5: ROC curve of Viola & Jones combined with Laptev ($H_2L$) compared with $H_2LS$

Table 1: Results of the different detectors (190 persons recognizable).

| Detector | Detected face |
|---|---|
| Haar Face | 37 (19.5%) |
| Haar Profil | 29 (15.3%) |
| Laptev | 140 (73.7%) |
| H$_2$L (Fusion) | 170 (89.5%) |

The next task is to reduce considerably the false positives (around 400 per panoramic, 40 per camera photo). When we take into account the skin tone the gain is significant, the number of false positives drops down to 50; we deleted 87.5% of false alarms. The global detection rate decreases a little bit, 86.2% cf Figure 5.

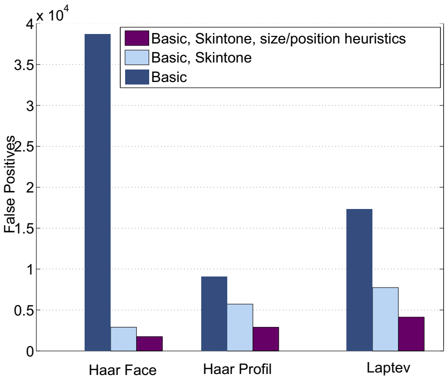

Figure 6: Evolution of false detections using the skin tone algorithm and position/size heuristics

Still remain to cancel around 50 false detections per panoramic. (Note that it is already a very good result, in our case the pictures are 20.7 MPx and usually detection algorithms are tested on webcams resolution, 320x240 = 150 000 px: more than 135 times less.)

As our system is fully calibrated (the geometry of our camera system is perfectly known) we can directly estimate that some pedestrian positions in the picture are impossible. Thus, we reduce the search space. Also, we know that the size of people faces cannot be more than a certain number of cm, some 100 px even if they are very close to the vehicle. We know that the probability to find people in the high part of the photo is very low: 0.001% of people are visible at their windows or balcony during the street navigation we did. Thus, it is reasonable to say that analyzing just the lower part of photos is enough (Anyway, for privacy preservation windows also need to be detected and occluded). And because many false alarms are in trees and windows, this help us to decrease to 20 false alarms per panoramic, i.e. 2 per photo (2.1 Mpixels).

The last thing to do is to find the most efficient way to hide faces. That means that faces must be unrecognizable and false alarms must be discreet. We tried different techniques among which blurring only the pixels classified as skin when they match a face detection area, and blurring all the face area where there is some skin inside. In fact, it seems more discreet for the human eye to have big regions of false alarms, than small non-homogeneous regions. So we apply on all the detected faces a very progressive gaussian blur with varying strength function of the face proximity to the cameras.

The quantity of data, 21 MPx per panoramic, 2.1 MPx per camera photo, makes the algorithms very time consuming. Ivan Laptev algorithm takes 4.7 minutes per camera shot, Viola & Jones 42 seconds for the face and profile classifiers, and 1.38 seconds for the SkinTone on a 2.4 GHz PentiumIV. The skin detection is a constraint on every detected face, hence we could imagine to search first for the skin pixels, then to launch the other algorithms on the windows surrounding the skin pixel detected. Actually, we tried such an approach but results were poor, the calculation time was not really reduced and we had a lot of different detections of the same face or no detection at all if we were reducing the parameter strength. This can be explained by the fact that boosting algorithms allow to concentrate very quickly on the windows with high probability to contain faces. Thus, the idea to target Adaboost algorithms on small region did not work out on our system. And because we have a lot of disparate skin pixels detected, we have to target many positions.

The final system gives 86.2% of good detection. 20 false alarms per panoramic. (seems a lot but means only 0.002 pix blurred by error).

## 5 Conclusion and future works

In this paper, we proposed a multi-boosting based approach to detect pedestrians in a street-level view panoramic system in order to blur their faces, using the combination of two boosting algorithms detecting faces [1] and bodies [2] with a skin tone detection algorithm we developped. The first objective was to detect a maximum of people then we tried to reduce the number of false alarms finding some heuristics and using the skin constraint. The results are quite encouraging for such unconstrained data: 86.2% of true positives and an average of 2 false positive detections per image (2.1 MPixels). This combination solution provides much more robust results than each detection algorithm performed independently. We ended up with an efficient system allowing us to stream images on our internet viewer. Further work will focus on increasing detection with new detectors combination, on false alarm reduction and on improving computational complexity.

## Acknowledgment

We would like to thank Ivan Laptev for providing us with his source code on Histogram of Oriented Gradients boosted for pedestrian detection.

## References

[1] P. Viola and M. Jones: "Rapid object detection using a boosted cascade of simple features" *CVPR 2001*

[2] Ivan Laptev: "Improvement of Object Detection Using Boosted Histograms" *Proc. BMVC'06 Edinburgh, UK*

[3] K. Levi and Y. Weiss: "Learning object detection from a small number of examples: the importance of good features" *CVPR 2004*

[4] N. Dalal and B. Triggs: "Histograms of oriented gradients for human detection" *CVPR 2005*

[5] P. Sabzmeydani and G. Mori: "Detecting Pedestrians by Learning Shapelet Features" *CVPR 2007*

[6] E. Seemann and M. Fritz and B. Schiele: "Towards Robust Pedestrian Detection in Crowded Image Sequences" *CVPR 2007*

[7] Shengye Yan and Shiguang Shan and Xilin Chen and Wen Gao: "Locally Assembled Binary (LAB) feature with feature-centric cascade for fast and accurate face detection" *CVPR 2008*

[8] D. Craciun and N. Paparoditis and F. Schmitt: "Automatic Pyramidal Intensity-based Laser Scan Matcher for 3D Modeling of Large Scale Unstructured Environments" *Fifth Canadian Conference on Computer and Robot Vision, pp. 18-25, 2008*

[9] K. Nishida and T. Kurita: "Pedestrian Detection by Boosting Soft-Margin SVM with Local Feature Selection" *MVA 2005 IAPR Conference on Machine Vision Applications*