Face Detection in a Pose Different than the One Learned

Pierre Bouges[1], Thierry Chateau[1], Christophe Blanc[1], Gaelle Loosli[2]

[1]Lasmea
CNRS UMR 6602 & Universite Blaise Pascal, 24 avenue des Landais, 63177 Aubiere Cedex, France

[2]Limos
CNRS UMR 6158 & Universite Blaise Pascal, 63173 Aubiere Cedex, France

pierre.bouges@univ-bpclermont.fr, thierry.chateau@univ-bpclermont.fr
christophe.blanc@univ-bpclermont.fr, gaelle@loosli.fr

Résumé – Détecter des visages de face est une tâche devenue commune. En revanche, détecter des visages qui ne sont pas de face reste problématique. Jusqu'à présent, détecter des visages dans une pose donnée nécessite de collecter énormément d'images de visage dans la pose souhaitée pour ensuite réaliser un apprentissage. Pour éviter cette collecte d'images fastidieuse et très consommatrice de temps, nous proposons une solution basée sur un détecteur de visage de face que nous modifions pour lui permettre de détecter des visages dans une pose différente. Pour cela, nous modifions la position des sous-fenêtres sur lesquelles s'appuie le détecteur. Les résultats obtenus montre une nette amélioration des performances.

Abstract – Detecting upright faces is now a common task. On the other hand, detecting non upright faces is still problematic. By now, detecting faces in a given pose require a huge number of training images of faces in the same pose. To avoid this tedious and time consuming collect, we propose a solution based on an upright face detector that we modify to allow it to detect faces in another pose. To do so, we modify all subwindows positions which are used at detection time. Results show a clear performance improvement.

1 Introduction

Since the nineties, the face detection problem is formulated as a binary classification problem. The goal is then to find a decision function which allows to distinguish face pattern from non-face pattern. The first significant results were obtained by Sung and Poggio [8] and by Rowley et al. [6]. Afterwards, most of the proposed solutions will rely on learning algorithms that are still commonly used nowadays : SVM and boosting-based algorithms. Osuna et al. [4] were the first to apply SVM to face detection. Papageorgiou et al. [5] also used a SVM and introduced haar features that will be extensively studied in literature. Finally, Viola and Jones [11] also use haar features and a cascade structure to rapidly detect faces.

All above solutions deal with the problem of upright face detection. In applications such that video surveillance, images usually contain faces with arbitrary rotation-off-plane (ROP) angles which make upright face detector to fail (the figure 1 exposes some faces with different ROP angles). Some solutions have been proposed to deal with these ROP ([7], [2]) but they are all based on the same idea : train several detectors (each one is in charge of a range of ROP angles) and combine them to make a decision. Although they prooved their efficiency, these methods are not really suitable for practical use due to the huge number of training example that must be collected. It is easy to find upright face databases on the web, but finding images of faces with ROP is a more challenging and time consuming task.

In this paper, we present a solution to detect faces with ROP without collecting training images of the corresponding pose. To do so, we train an upright face detector and we modify it to allow face detection in a different pose.

The rest of this paper is organized as follow : in section 2, we present the upright face detector that we use. Then, the section 3 introduces the solution that we propose. Some experiments are carried out in section 4 and finally, a conclusion is given in section 5.

(a) Rotation of 90 ° (b) Rotation of 67,5 ° (c) Rotation of 45 ° (d) Rotation of 22,5 °

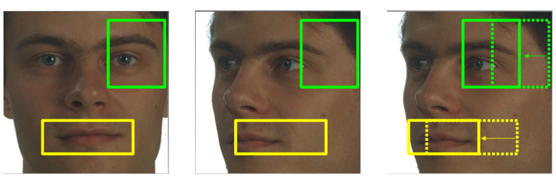
Fig. 1: ROP example.The figures (a), (b), (c) and (d) show some rotations around y-axis.

2 Upright faces detection

Basically, our solution is made up of an upright face detector that we modify to be able to detect faces with a given ROP angle. So far, the best results on face detection were reported by solutions based on boosting algorithms. That's why we decide to use one of them. We chose the detector proposed by Tuzel et al. [10] for its good performance. This one uses covariance matrix [9] as features and Logitboost [1] as learning algorithm. The final detector is a cascade of logitboost classifier and each logitboost classifier is made up of several weak classifiers (linear regression function here). During detection time, the detector is applied to a set of detection window R. At a given cascade level, each weak classifier's f_i goal is to classify a subwindow r_i of the detection window R.

3 Detecting faces in a different pose

To incorporate the fact that faces are not upright, we propose to adjust all subwindows' positions. Our idea is illustrated in the figure 2.



(a) (b) (c)

FIG. 2: The figure 2(a) shows two interesting subwindows used to detect upright faces. In figure 2(b), we represent the same subwindows on a face with a ROP angle of 45 °. The two subwindows are no more interesting. To alleviate this problem, we can modify the position of the two subwindows as showed in figure 2(c). Note that the position modification can lead to a modification of the subwindow size (see the yellow subwindow for example).

To modify a subwindow position, we propose to use the 3D transformation which exists between an upright face and the same face in another pose. In our case, these transformations are the set of rotations around x-axis and y-axis. To simulate a rotation, we need a 3D face model. Building an accurate 3D face model require at least two images per face. As our will is to avoid gathering images ohter than upright faces, we decide to represent a face with the simplest model : an ellipsoid. The idea is then to place each subwindow on the ellipsoid, turn the ellipsoid and finally get back all new subwindows positions. Let us consider a point $p_1^i = (u_1\ v_1)^T$ of an image of size $w \times w$[1] whose coordinates are expressed in the image coordinate system $\mathcal{R}_i$. The process to compute the position of this point after a rotation defined by θ_x and θ_y[2] angle is made up of the three following steps :

1. We associate a point P_1^i to the point p_1^i. This point belongs to the ellipsoid and we just have to compute the z-coordinate w_1 with the help of the ellipsoid equation expressed in $\mathcal{R}_i$:
$$\frac{(u-u_0)^2}{a^2} + \frac{(v-v_0)^2}{b^2} + \frac{(w-w_0)^2}{c^2} = 1 \quad (1)$$
where $u_o = w/2$, $v_o = w/2$ and $w_o = 0$.
2. We express P_1^i in the coordinate system $\mathcal{R}_e$ whose origin is the ellipsoid center[3]. This gives us the P_1^e point :
$$\begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \\ \tilde{d}_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -w/2 \\ 0 & 1 & 0 & -w/2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ v_1 \\ w_1 \\ 1 \end{bmatrix} \quad (2)$$
and then, we have $P_1^e = (\tilde{x}_1/\tilde{d}_1\ \tilde{y}_1/\tilde{d}_1\ \tilde{z}_1/\tilde{d}_1)^T = (x_1\ y_1\ z_1)^T$ at which we apply the rotation to obtain the P_2^e point :
$$P_2^e = (x_2\ y_2\ z_2)^T = R_y(\theta_y) \times R_x(\theta_x) \times P_1^e \quad (3)$$
where $R_y(\theta_y)$ and $R_x(\theta_x)$ are rotation matrices around y-axis and x-axis.
3. Finally, we express P_2^e in $\mathcal{R}_i$ to get the P_2^i point :
$$\begin{bmatrix} \tilde{u}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \\ \tilde{d}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -w/2 \\ 0 & 1 & 0 & -w/2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} \quad (4)$$
We have $P_2^i = (\tilde{u}_2/\tilde{d}_2\ \tilde{v}_2/\tilde{d}_2\ \tilde{w}_2/\tilde{d}_2)^T = (u_2\ v_2\ w_2)^T$. The point we are looking for is $p_2^i = (u_2\ v_2)^T$.

To know the position of a subwindow r_i after a rotation, we apply the above process to the four corners of r_i.

4 Experiments

In this section, we train an upright face detector and we then apply our modification. The resulting system is tested on the FERET database.

4.1 Train of the upright face detector

The upright face detector was trained with the Labeled Faces in the Wild database [3] which consists of 13233 upright faces. Negative samples were generated from 1310 images containing no faces. Each sample size is 24×24. Each cascade level is trained to correctly detect at least 99.8% of the positive samples while rejecting 40% of the negative samples. Moreover, 4000 positve samples and 8000 negative samples are used per level. The final detector is made up of 5 levels.

[1] we consider a square image because the detection windows are square

[2] θ_x (resp. θ_y) represents the angle of the rotation around the x-axis (resp. y-axis)

[3] it also corresponds to the image center

4.2 System performance

The FERET database provides faces with different angles with respect to the y-axis. We evaluate our system on three different angles : 22.5 °, 45 °and 67.5 °. For each angle $\theta \in \{22.5,\ 45,\ 67.5\}$, we expose three ROC curves :

1. a curve " up detector + upright face " which corresponds to the use of the upright face detector on images containing upright faces ;

2. a curve " up detector + face turned θ°left " which shows the result of the use of the upright face detector on images containing the faces of the same people but turned of θ°on the left ;

3. a curve " up detector + face turned θ°left + modification " which corresponds to the use of the modified upright face detector (all subwindows positions are modified before applying the detector) on images containing the faces of the same people but turned of θ°on the left ;

For $\theta = 45°$(see figure 4), we also plot another curve : " spe detector + face turned θ°left ". To produce this curve, we have trained a detector with images of faces turned of 45°on the left.
All other results are showed on figure 3. These results allow us to say that our approach is really beneficial in the case of the detection of faces that are not upright. In the best case, detection rate is higher of 40% (see figure 4). With the 67.5°angle, some subwindows disappear after the position modification (it occurs when the new position is on the other side of the ellipsoid). To alleviate this problem, a heuristics is used[4] and results are showed on figure 3(b).

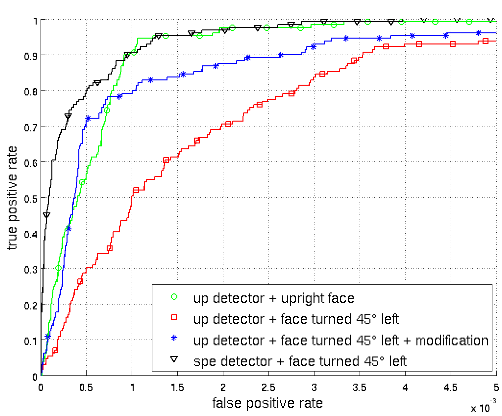


FIG. 4: ROC curves on upright faces and faces turned of 45°left.

4.3 Ellipsoid parameters

Here, we present some results on parameters that can affect our method. The set of parameters was tested with the modified uprigt face detector on faces turned of 45°on the left.
First, we study the influence of the window used to modify the subwindow position. In section 3, we explain that we consider an image of size $w \times w$. The figure 5 show some results for different value of the w parameter. The best AUC[5] is obtained for $w = 48$.

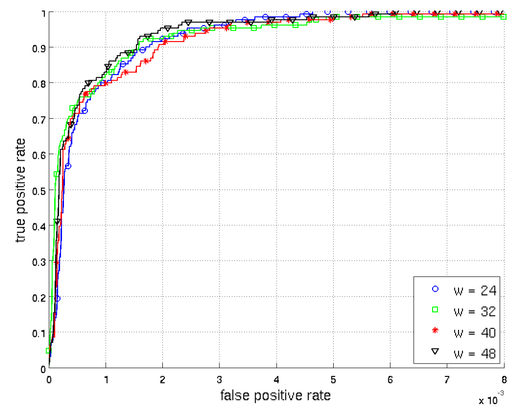


FIG. 5: Comparison of different window size to modify the subwindow position.

We also test the influence of the three ellipsoid radii a, b and c. Each time, two of them equal $w/2$ and the last one varies in a range of values. Experiments show that b and c don't modify the performance. Only a has an impact (see figure 6). Setting $a = 0.85 * w/2$, $b = w/2$ and $c = w/2$ give the best results.

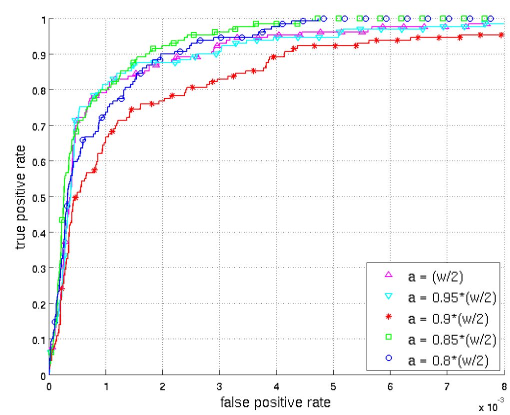
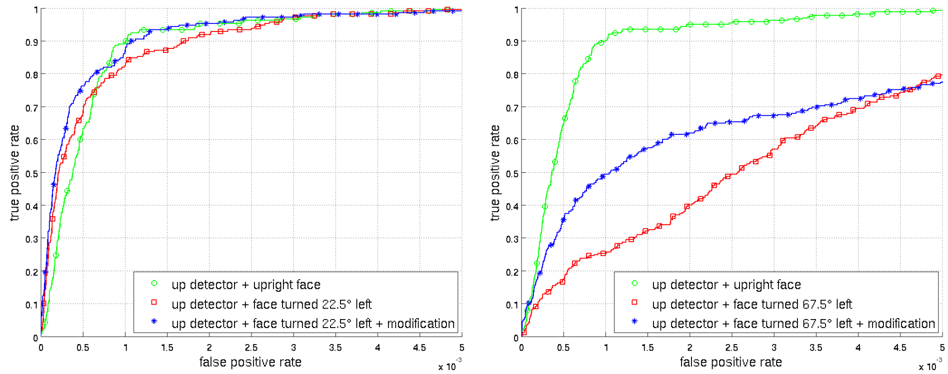


FIG. 6: Comparison of different radii along the x-axis (noted a on the ROC legend). The radii along y and z-axis are kept equal to $w/2$. We first test a equal to $w/2$ and then, a lower than $w/2$.

[4] As a face is pretty much symmetric with respect to the y-axis, we consider the symetric subwindow

[5] **A**rea **U**nder the ROC **C**urve is a criteria to compare ROC curves. The higher the AUC is, the better the ROC curve is.



(a) Rotation of 22,5 °

(b) Rotation of 67,5 °

FIG. 3: ROC curves of upright faces detection, turned faces detection and turned faces detection with the proposed correction.

5 Conclusion

In this paper, we present a solution to detect faces in a different pose than the one that was leanerd. Our method can be easily implemented and incorporated into an existing face detection system. The only constraint is the following : when a detection window R must be classified as face or not, the classifier must use several subwindow r_i of R to make a decision[6]. Our will is to be able to use an upright face detector, which is easy to learn, on faces that are not upright. To do so, we propose to modify all subwindows positions to incorporate the face pose into the existing detector. Experiments show that our approach can greatly improve detection rate. The future work is to extend the method to be able to deal with important pose modification. In that case, some subwindows are no more usable which lead to missing values at prediction time. That's why we plan to study methods that can handle these missing values.

Acknowledgements

We want to thank OSEO for supporting us. Our work is part of the Biorafale project which aims at detecting and recognizing dangerous fans.

References

[1] J. Friedman, T. Hastie and R. Tibshira. *Additive Logistic Regression : a Statistical View of Boosting.* Annals of Statistics, vol.28, no. 2, p. 337-407, 2000.

[2] C. Huang, H. Ai, Y. Li and S. Lao. *High-Performance Rotation Invariant Multiview Face Detection.* in IEEE Transactions on Pattern Analysis and Machine Intellignce (PAMI), vol. 29, no. 4, Avril 2007.

[3] G. B. Huang, M. Ramesh, T. Berg and E. Learned-Miller. *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments.* University of Massachusetts, Amherst, Technical Report 07-49, October 2007.

[4] E. Osuna, R. Freund et F. Girosi. *Training Support Vector Machines : An Application to Face Detection.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p. 130-136, 1997.

[5] C. Papageorgiou, M. Oren et T. Poggio. *A General Framework for Object Detection.* In International Conference on Computer Vision (ICCV), 1998.

[6] H. Rowley, S. Baluja et T. Kanade. *Human Face Detection in Visual Scenes.* Technical Report CMU-CS-95-158R, universitede Carnegie Mellon, Novembre 1995.

[7] H. Schneiderman et T. Kanade. *A Statistical Method for 3D Object Detection Applied to Faces and Cars.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, p. 746-751, 2000.

[8] K. Sung et T. Poggio. *Example-based Learning for View-based Human Face Detection.* A.I Memo 1521, MIT A.I. Lab, Decembre 1994.

[9] O. Tuzel, F. Porikli and P. Meer, *Region Covariance : A Fast Descriptor for Detection and Classification.* In European Conference on Computer Vision (ECCV), 2, p. 589-600, 2006.

[10] O. Tuzel, F. Porikli et P. Meer. *Human Detection via Classification on Riemannian Manifolds.* In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2007.

[11] P. Viola et M. Jones. *Rapid Object Detection Using a Boosted Cascade of Simple Features.* In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, p. 511-518, 2001.

[6] This is typically the case of all recent boosting-based method.