Apprentissage hiérarchique pour la détection de visages

F. Fleuret D. Geman

Projet IMEDIA, INRIA Rocquencourt

Domaine de Voluceau BP 105 78153 Le Chesnay Cedex
{francois.fleuret, donald.geman}@inria.fr

Résumé

Notre but est de détecter toutes les instances d'un objet générique dans une image en niveaux de gris. On estime l'efficacité à partir du nombre de fausses alarmes et du coût algorithmique nécessaire pour ne rater aucune détection. A partir d'une base d'images d'apprentissage, nous construisons récursivement une familles d'arrangements de bords ("patrons hiérarchiques"). Ces arrangements sont "décomposables" : chacun d'entre eux peut être divisé en deux sous-arrangements fortement corrélés, qui peuvent à leur tour être décomposés, etc. La détection est basée sur la recherche d'un nombre suffisant d'arrangements de chaque taille. En décomposant également l'espace des poses, la détection globale est finalement hiérarchique aussi bien dans l'exploration des poses que dans la représentation des visages.

Mots Clef

Détection de visage, hiérarchique, corrélation, apprentissage statistique

Abstract

Our goal is to detect all instances of a generic object class such as a face in greyscale scenes. Performance is measured by the number of false alarms and the amount of computation necessary for no missed detections. Starting from training examples, we recursively learn a hierarchy of increasingly complex spatial arrangements of edge fragments ("coarse-to-fine templates"). The arrangements are "decomposable": Each can be split into two correlated subarrangements, each of which can be further divided, etc. Detection is based on finding a sufficient number of arrangements of various sizes. By also decomposing the space of poses, the final search is coarse-to-fine in both the exploration of poses and the representation of faces.

Keywords

Face detection, coarse-to-fine, correlation, statistical learning

1 Introduction

Partant d'un ensemble d'apprentissage contenant des exemples d'un objet générique (par exemple des visages, cf. figure 1), notre but est de construire un algorithme pour détecter et localiser toutes les instances de cet objet dans des images en niveaux de gris. Les exemples de la base d'apprentissage sont des sous-images contenant une unique instance de l'objet dans une position quelconque, par exemples des vues frontales de visages à différentes échelles, différentes inclinaisons, etc. Même si les fonds des images de la base d'apprentissage sont très simples, l'algorithme doit pouvoir fonctionner dans des scènes naturelles très complexes. Nous estimons les performances de l'algorithme à l'aide du taux de faux positifs et du coût algorithmique nécessaires pour obtenir un taux de faux négatifs très faible, en utilisant une base d'apprentissage de taille réduite.

Ce travail s'inscrit dans un projet plus large sur la reconnaissance d'images vue comme un exemple du "jeu des vingt questions". Nous construisons les fonctions de l'image et la stratégie d'exploration en même temps, et d'une manière très hiérarchisée aussi bien pour la représentation de l'objet que pour son positionnement dans l'espace des poses. Ce paradigme a été analysé dans le contexte des arbres de décision et de la réduction de l'incertitude pas à pas dans [1], [4], [5] et [11]. Bien que l'approche de l'apprentissage soit différente ici, et qu'il n'y ait pas de construction d'arbres proprement dite, l'algorithme final peut être mis sous la forme d'un énorme arbre binaire récursif, dont les questions sont basées sur des comparaisons entre les niveaux de gris des pixels.

Le détecteur que nous proposons ici a la forme d'une succession de *détecteurs dédiés* à des sous-ensembles de

FIG. 1 – *Quelques une des* 300 *images de la base d'apprentissage, extraites de la base de données "ORL Database of Faces".*

plus en plus contraints de l'ensemble d'apprentissage total. Il s'agit ici de contrainte sur la pose du visage à détecter (position des yeux et de la bouche). Finalement, la forme la plus globale du détecteur correspond donc à un partitionnement de l'espace des poses. Tout ces détecteurs dédiés sont construits de la même manière; seul l'ensemble d'images utilisé pour l'apprentissage diffère.

Pour chacun de ces détecteurs, nous construisons une succession de tests de plus en plus complexes. La détection se fait en s'assurant successivement de la présence d'un nombre minimum de tests de chaque complexité. Comme dans [2], les tests sont des fonctions booléennes de l'image correspondant à la présence ou l'absence d'arrangements de fragments de bords. Ces fragments ont une localisation et une orientation approximative; la définition est volontairement tolérante pour être invariante par des transformations géométriques. Les arrangements n'ont pas de signification ou d'interprétation géométrique; à la place nous voulons simplement qu'ils soient le plus probables possible sur les objets recherchés.

Nous introduisons la notion "d'arrangement décomposable" et un algorithme pour construire un grand nombre de tels arrangements à partir de l'ensemble d'apprentissage. "Décomposable" signifie que l'arrangement peut être mis sous la forme d'une conjonction de deux sous-arrangements très corrélés, qui peuvent à leur tour être décomposés en plus petits arrangements très corrélés, etc. jusqu'aux fragments de bords eux-mêmes. L'algorithme de construction est une procédure qui permet de construire des arrangements de plus en plus complexes de manière récursive. La motivation est que la probabilité qu'un arrangement de taille k apparaisse sur une instance de l'objet décroît très progressivement quand k croît, nous assurant donc que les tests seront très discriminants pour séparer les objets du fond. Cette hypothèse sera justifiée théoriquement et illustrée empiriquement.

L'expérimentation que nous décrivons consiste à détecter des vues frontales de visages. Nous utilisons la base de données ORL contenant 300 visages (10 vues de 30 personnes) ce qui est un ensemble assez réduit, mais suffisant dans notre cas puisque nous ne faisons qu'estimer des corrélations. L'apprentissage ne prend d'ailleurs que quelques minutes sur un PC. En particulier nous n'estimons pas de grands ensembles de paramètres comme dans les autres systèmes d'apprentissage statistique.

2 Détection d'objet invariante

La détection des instances d'un objet générique sans utilisation d'informations de couleur, profondeur ou mouvement a été énormément étudiée dans la littérature informatique. Dans le cas des visages, une multitude de méthodes a été proposée, par exemple les réseaux de neurones artificiels [8], [9], les "support vector machines" [7], la mise en correspondance de graphes [6], l'inférence Bayésienne [3], les modèles déformables [12], et les précurseurs de notre méthode qui ont déjà été cités.

Une des principales difficultés réside dans la variation de l'apparence des visages due aux variations d'éclairage; voir par exemple la discussion dans [10]. Notre approche consiste à introduire une grande invariance photo-métrique dans la définition des fragments de bords en ne considérant que des comparaisons entre des différences d'intensités (cf. section 4). De même notre approche de l'invariance géométrique est assez explicite. Chaque arrangement est une conjonction de tests élémentaires, qui sont eux-mêmes des disjonctions de morceaux de bords; on compare finalement le nombre d'arrangements présents à un seuil. La détection est finalement basée sur une gigantesque disjonction de conjonctions de fragments de bords non localisés, ce qui correspond à une gestion très explicite de l'invariance géométrique, et est très différent de la majorité des autres approches; voir [2] pour une discussion plus complète.

3 Cadre statistique

Nous formalisons une tâche simple de détection dans une sous-image extraite de la scène complète. Cette opération se réduit donc à la classification des sous-images en deux classes: "visage" / "fond", et correspond à l'opération faite par un des détecteur dédié.

Soit $\mathcal{I}$ l'ensemble des sous-images $I = \{I(x), x \in R\}$, où R est la grille de référence (par exemple 32×32) et $I(c)$ est normalisé de manière standard par exemple en 256 niveaux de gris. Les sous-images sont réparties en deux catégories: "objet" et "fond". Les images d'objets ne contiennent qu'une seule instance de la classe de l'objet, dans une pose de référence, ce qui veut dire que l'objet est grossièrement centré dans la sous-image et que sa taille est à peu près celle de la grille R (en choisissant les images de l'objet de manière à contraindre la pose, on peut construire un détecteur dédié à un sous-ensemble de poses). Les images de fond sont toutes les autres. Soit $Y(I) \in \{0, 1\}$ la classe de I, où "0" correspond à la classe "fond".

Notons P une mesure de probabilité sur $\mathcal{I}$. Nous pouvons voir P comme la mesure empirique de toutes les sous-images de toutes les images d'une grande base de données. Alors P induit deux mesures conditionnelles: $P_0(.) = P(.|Y = 0)$ la distribution sur la classe "fond", et $P_1(.) = P(.|Y = 1)$, la distribution sur la classe objet. Etant donné un classificateur $f : \mathcal{I} \to \{0, 1\}$, l'erreur de faux négatif est $\alpha(f) = P_1(f = 0)$ et l'erreur de faux positif est $\beta(f) = P_0(f = 1)$. Idéalement, on cherche à minimiser $\beta(f)$ avec la contrainte $\alpha(f) = 0$. Nous considérons que l'ensemble d'apprentissage $\mathcal{L}$ est un échantillon aléatoire de $\mathcal{I}$ sous P_1. Nous devons tenir compte du fait que la taille de cette échantillon n'est pas suffisante pour estimer un grand nombre de paramètres interdépendants. Nous utilisons des exemples négatifs (c'est à dire sous P_0) pour estimer le taux de fausses alarmes, mais pas pour construire le classificateur.

Comme notre approche est inductive plutôt que déductive, nous ne proposons pas de modèle ni pour P_0, ni pour P_1 ; à la place, nous nous reposons sur la mesure empirique $\hat{P}_1$ estimée à partir de $\mathcal{L}$. Donc, nous construisons directement un classificateur à partir de $\mathcal{L}$. L'apprentissage se ramène finalement à estimer la probabilité P_1 d'évènements de $\mathcal{I}$; ces probabilités déterminent les composants du classificateur et sont déduites des fréquences relatives dans $\mathcal{L}$.

4 Une hiérarchie de tests

Partant de tests primitifs, localisés et fonctions de quelques pixels, nous cherchons à construire progressivement une hiérarchie de tests de plus en plus complexes, pour arriver à des tests globaux et denses dans l'image, dont les statistiques dans les deux populations deviennent de plus en plus contrastées. L'efficacité algorithmique est obtenue en cherchant des structures de plus en plus complexes afin de séparer les instances des objets de celles du fond. Une grande partie des sous-images peut être facilement éliminée des candidats objets en ne considérant que des tests très primitifs, par exemple simplement en comptant le nombre de bords présents; des confusions plus globales nécessitent la considération de corrélations d'ordres plus élevés et de la dépendance à long terme.

4.1 Tests élémentaires

Tous nos tests sont des conjonctions de *tests élémentaires binaires*. Ces derniers sont des disjonctions de filtre locaux. Dans nos expériences, les filtres locaux détectent la présence ou l'absence de fragments de bords; d'autres filtres, plus sophistiqués, pourraient être plus efficaces. Chaque filtre est appliqué à toutes les positions de R, et possède une orientation (horizontale ou verticale) et une polarité (positive ou négative). Considérons un bord horizontal, de polarité positive, situé en x. Cela signifie que $I(x) > I(x')$, où x et x' sont des voisins verticaux, et que $I(x) - I(x') > \max_y\{|I(x)-I(y)|, |I(x')-I(y)|\}$ pour un certain nombre de pixels voisins y. Sur la figure 2 nous montrons un visage provenant de la base d'apprentissage (image du haut) et les fragments de bords détectés (image du bas).

Il y a un test élémentaire $X_i = X_i(I)$ pour chaque fragment de bord $i = 1, 2, \ldots, d$, où $d \approx 4|R|$. Le test $X_i = 1$ si le bord correspondant est présent dans un petit voisinage de x_i et $X_i = 0$ sinon. La taille du voisinage (le degré de "tolérance") dépend du sous-ensemble de poses de l'ensemble d'apprentissage; il est choisi de manière à ce que la probabilité des tests élémentaires soit de l'ordre de $\frac{1}{2}$.

Pour améliorer le taux de détection en éliminant des fausses alarmes, on utilise également des détecteurs construits à partir de tests élémentaires "négatif". Un tel test $X_i^\star = 1$ si le bord correspondant n'est pas présent dans le voisinage.

4.2 Arrangements décomposables

Nous appelons *arrangement* un sous-ensemble

$$A \subset \{1, \ldots, d\}$$

la variable aléatoire correspondante

$$X_A(I) = \prod_{i \in A} X_i(I)$$

sur $\mathcal{I}$ est simplement la conjonction spatiale de tests élémentaires: $X_A = 1$ si et seulement si $X_i = 1$ pour tous les $i \in A$. Soit $suppX_i \subset R$ l'ensemble des pixels qui apparaissent dans la définition de X_i. Pour limiter la taille de la famille des arrangements, nous faisons l'hypothèse que $suppX_i \bigcap suppX_j = \emptyset$ pour tout $i \neq j$. Nous notons $|A|$ le cardinal de A. Cet ensemble

constitue notre famille de tests ; le classificateur sera construit à l'aide d'un sous-ensemble de ces tests.
Nous voulons trouver des arrangements A ayant des statistiques les plus différentes possibles pour P_0 et pour P_1. Parceque l'estimation de la probabilité sous P_0 est problématique, nous essayons d'obtenir cette disparité en construisant de grands arrangements vraisemblables sous P_1. La taille seule les rend rare sous P_0. La construction est basée sur la corrélation. Notons $\rho(U,V)$ le coefficient de corrélation de deux variables aléatoires U et V sous P_1. Pour des variables aléatoires booléennes telles que

$$0 < P_1(U=1), P_1(V=1) < 1$$

nous avons :

$$\rho(U,V) = \frac{P_1(U=1,V=1) - P_1(U=1)P_1(V=1)}{(P(U=1)P(U=0)P(V=1)P(V=0))^{1/2}}$$

Considérons les arrangements X_iX_j de taille deux. Nous pouvons filtrer de telles paires à l'aide de la contrainte

$$\rho(X_i, X_j) \geq \rho$$

pour un certain seuil ρ, $0 < \rho < 1$. Nous sélectionnons ainsi les paires de tests qui ont tendance à apparaître (respectivement ne pas apparaître) simultanément sur les objets. De même $X_iX_jX_k$ serait un bon candidat pour un arrangement discriminant de taille trois, si $\rho(X_iX_j, X_k) \geq \rho$. En continuant ainsi, nous pouvont isoler de bon candidats de taille quatre en combinant deux "bonnes" paires de tailles deux qui vérifient $\rho(X_iX_j, X_kX_l) \geq \rho$. Etc.
Nous définissons une *décomposition* de A comme une succession de partitions binaires emboitées, jusqu'à des singletons sous-ensembles de $\{1,\ldots,d\}$. Nous faisons également l'hypothèse que chaque partition d'un sous-ensemble le divise en deux partie égales si l'ensemble est de cardinal pair, et en deux parties de cardinaux différents de un si l'ensemble est de cardinal impair. Nous dirons qu'il s'agit d'une *ρ-décomposition* si l'inégalité sur la corrélation est vérifiée pour chaque split. Sur la figure 3 nous montrons une telle décomposition de $A = \{1,2,4,5,9\}$. C'est une ρ-décomposition si :

- $\rho(X_1X_4,\ X_2X_5X_9) \geq \rho$
- $\rho(X_1,\ X_4) \geq \rho$
- $\rho(X_5X_9,\ X_2) \geq \rho$
- $\rho(X_5,\ X_9) \geq \rho$

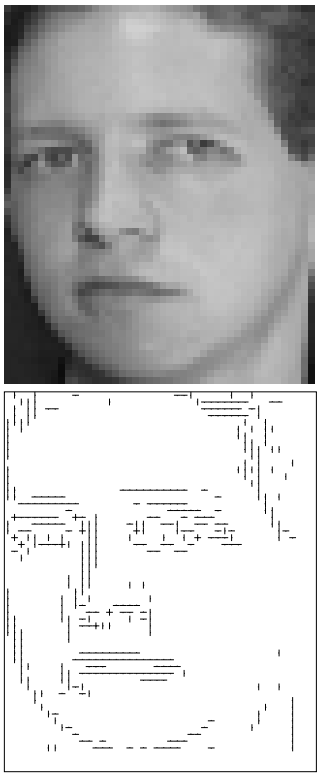

FIG. 2 – *Haut: Un visage provenant de la base d'apprentissage. Bas : La carte des bords.*

Finalement, un arrangement A, ou la variable aléatoire correspondante X_A, sera dite *ρ-décomposable* s'il existe au moins une ρ-décomposition de A. Pour résumer :

Définition: *Un arrangement A est ρ-décomposable si c'est un arrangement de taille un (un test élémentaire) ou bien s'il existe deux arrangements ρ-décomposables B et C avec :*

- $A = B \cup C, \quad B \cap C = \emptyset$
- $||B| - |C|| \leq 1$
- $\rho(X_B, X_C) \geq \rho$

En général $P_0(X_A = 1)$ et $P_1(X_A = 1)$ dépendent de A et décroissent quand $|A|$ augmente. Une hypothèse raisonnable pour P_0 est une décroissance exponentielle, et c'est ce que nous observons effectivement dans nos résultats expérimentaux. Par contre, si A est un arrangement ρ-décomposable, nous pouvons nous attendre à une décroissance moins rapide sous P_1. En fait on peut prouver que la vitesse de décroissance est au pire de la forme $\rho^{\log_2 k}$. La conséquence est que pour pour des valeurs "raisonnables" de ρ, on a

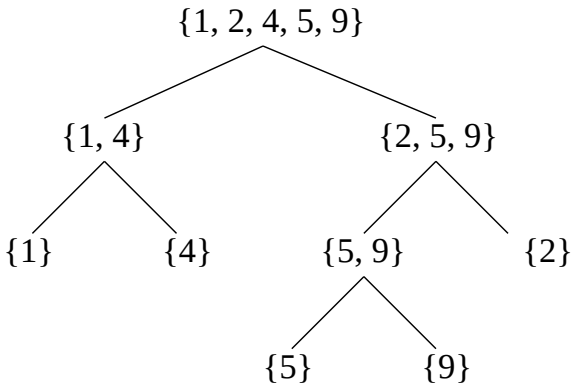

FIG. 3 – *Un arrangement ρ-décomposable peut être décomposé récursivement en conjonctions d'arrangements corrélés et à peu près de même tailles.*

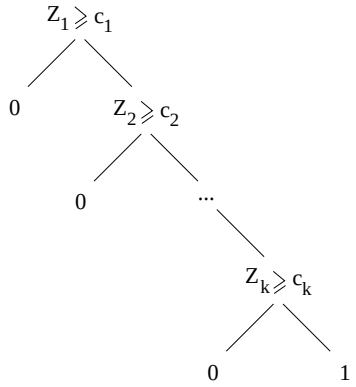

FIG. 4 – *Le détecteur final est une séquence hiérarchisée de filtres de plus en plus complexes.*

$P_1(X_A = 1) \gg P_0(X_A = 1)$ pour de "grand" A. On ne peut néanmoins rien dire pour le rapport des vraisemblances puisque nous ne proposons aucun modèle pour P_0. Pour $P_1(X_A = 1)$ par contre nous pouvons calculer une borne inférieure précise. Notons $\mathcal{A}(k, \rho)$ l'ensemble des arrangements de taille k qui sont ρ-décomposables. On peut montrer le théorème suivant :

Théorème: *Pour tout $k \geq 1$, $\rho > 0$ et $A \in \mathcal{A}(k, \rho)$,*

$$P_1(X_A = 1) \geq \min_{1 \leq i \leq d} P_1(X_i = 1) \cdot \rho^{\log_2 k}. \qquad (1)$$

5 Structure d'un détecteur dédié à un sous-ensemble de poses

La détection peut être vue comme une séquence de détecteurs de complexités croissantes. Un test n'est utilisé que si tous les tests plus simples ont rejeté l'hypothèse "fond". Comme la très grande majorité des images à tester sont des images de "fond", l'algorithme est finalement un processus très hiérarchisé avec lequel seules quelques images sont examinées en détail.
Fixons ρ. Chaque détecteur est basé sur le nombre $Z_{k,\rho}$ d'arrangements ρ-décomposables de taille k présent sur l'image I :

$$Z_{k,\rho}(I) = \sum_{A \in \mathcal{A}(k,\rho)} X_A(I)$$

Soit K le plus grand k tels que les arrangements de taille k "couvrent" la classe objet, c'est à dire

$$P_1(Z_{k,\rho} \geq 1) = 1$$

(au cours de nos expériences, il n'est jamais arrivé que les arrangements de taille k couvrent la classe objet mais pas les arrangements de taille $j < k$). Etant donnés des seuils $\{c_1, \ldots, c_K\}$, nous classifions une image I comme un objet si elle contient plus de c_k arrangements ρ-décomposables pour tous les $k = 1, \ldots, K$. Ce qui revient à dire que le classificateur se met sous la forme :

$$f_\rho(I) = \prod_{k=1}^{K} 1_{\{Z_{k,\rho}(I) \geq c_k\}}$$

Les seuils $c_1, \ldots, c_K$ sont définis par

$$c_k = \max\{j : P_1(Z_{k,\rho} \geq j) = 1\}$$

Ils correspondent donc aux valeurs maximum qui permettent d'obtenir $\alpha(f_\rho) = 0$. Finalement, nous implémentons f_ρ sous la forme d'une succession de tests comme représenté sur la figure 4.
Finalement, le choix le plus naturel pour ρ correspond à la valeur qui minimise l'erreur de faux positifs :

$$\rho^* = \arg\min_{\rho} \beta(f_\rho)$$

Nous n'avons pas procédé à une détermination systématique de la valeur ρ et nous prenons une valeur constante : $\rho = 0.1$.

6 Apprentissage

En pratique nous ne pouvons pas construire directement f_ρ parce-que nous ne disposons pas des ensembles $\mathcal{A}_{k,\rho}, k = 1, \ldots, K$. Leur construction nécessite la connaissance de P_1 et il y en a un trop grand nombre.
A la place, nos expériences sont basées sur une approximation. Comme nous ne pouvons pas construire *tous* les arrangements ρ-décomposables, nous essayons de déterminer un nombre fini n d'entre eux pour chaque complexité $k \leq K$. Donc, étant donné $\mathcal{L}$, l'un des objectifs de l'apprentissage est d'estimer une sous famille

de $\mathcal{A}_{\mathcal{L}}(k,\rho) \subset \mathcal{A}(k,\rho)$ de taille n pour tout $k \leq K$. L'autre objectif est d'estimer les seuils $c_1, \ldots, c_K$.
Bien que la définition d'un arrangement décomposable soit descendante, la construction proprement dite est ascendante. Les corrélations sont estimées à l'aide de $\hat{P}_1$, mesure empirique déduite de $\mathcal{L}$.
La construction est récursive : nous construisons d'abord une famille $\{X_i X_j\}$, puis une famille $\{X_i X_j X_k\}$, etc. En fait, pour construire $\mathcal{A}_{\mathcal{L}}(2k,\rho)$ nous n'avons besoin que de $\mathcal{A}_{\mathcal{L}}(k,\rho)$; et pour $\mathcal{A}_{\mathcal{L}}(2k+1,\rho)$ que de $\mathcal{A}_{\mathcal{L}}(k,\rho)$ et $\mathcal{A}_{\mathcal{L}}(k+1,\rho)$.
Considérons le cas pair. Soit $\mathcal{A}'_{\mathcal{L}}(2k,\rho)$ l'ensemble de tous les arrangements $A_1 \bigcup A_2$ où

- $A_1, A_2 \in \mathcal{A}_{\mathcal{L}}(k,\rho)$;
- $\hat{\rho}(X_{A_1}, X_{A_2}) \geq \rho$;
- $suppX_{A_1} \bigcap suppX_{A_2} = \emptyset$.

Nous voulons sélectionner un sous-ensemble $\mathcal{A}'_{\mathcal{L}}(2k,\rho)$, de taille n si possible. Généralement $n \ll |\mathcal{A}'_{\mathcal{L}}(2k,\rho)| \ll n^2$. Si $|\mathcal{A}'_{\mathcal{L}}(2k,\rho)| \leq n$, alors $\mathcal{A}_{\mathcal{L}}(2k,\rho) = \mathcal{A}'_{\mathcal{L}}(2k,\rho)$. Sinon $\mathcal{A}_{\mathcal{L}}(2k,\rho)$ est construit en échantillonnant aléatoirement $\mathcal{A}'_{\mathcal{L}}(2k,\rho)$, en prenant les arrangements un par un parmi ceux qui couvrent les images de l'ensemble d'apprentissage qui sont couvertes par le plus petit nombre d'arrangements sélectionnés jusque là. En d'autres mots, cet échantillons aléatoire est construit pour maximiser

$$\min_{\omega \in \mathcal{L}} \left\{ \sum_{A \in \mathcal{A}_{\mathcal{L}}(2k,\rho)} X_A(\omega) \right\}$$

Le processus de construction est initialisé en considérant les n arrangements de tailles un qui sont les plus fréquents (c'est à dire que nous choisissons parmis les tests élémentaires, tels qu'ils sont définis dans 4.1, les n qui ont les probabilités marginales les plus élevées), et il se termine quand on rencontre le premier k tel qu'on ne peut pas couvrir les images d'objets à l'aide d'arrangements de taille k. Finalement, les estimateurs naturels des $c_1, \ldots, c_K$ sont de la forme :

$$\hat{c}_k = max \left\{ c : \hat{P}_1 \left(\sum_{A \in \mathcal{A}_{\mathcal{L}}(\rho,k)} X_A \geq c \right) = 1 \right\}$$

Cette définition surestime de manière évidente les c_k, en fait cette surestimation peut être critique dans certains cas. Pour éviter cela, nous ne prenons que la moitié de cette valeur dans toutes nos expériences ; cela nous permet d'obtenir $\alpha(f) = 0$, contrainte fondamentale dans notre approche du problème.

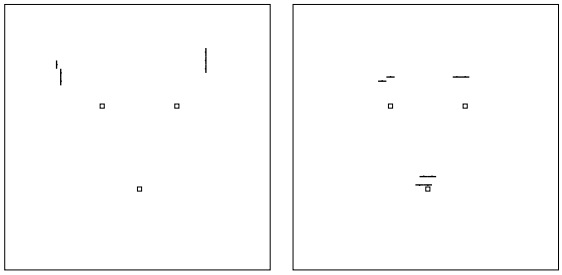

FIG. 5 – *Exemples d'arrangements de tailles* 6 *et* 8.

7 Résultats sur des images normalisées

L'ensemble d'apprentissage $\mathcal{L}$ est construit à partir de la base de données de visages ORL qui contient 400 images, correspondant à 10 images de chacun des 40 personnes représentées. Après sous-échantillonnage, nous disposons de 400 images au format 46×56, et nous marquons pour chacune d'entre elles les positions des yeux et de la bouche ; il y a relativement plus de variation pour l'échelle (distance entre les yeux) et l'élongation (rapport entre la distance entre les yeux et la distance entre la bouche et le point entre les yeux) qu'en orientation. Nous utilisons 300 images (de 30 personnes) pour l'apprentissage, et le reste pour estimer l'erreur de faux négatif.

La figure 5 montre deux arrangements qui ont été appris à partir de $\mathcal{L}$, l'un de taille six, l'autre de taille huit. L'orientation et la polarité sont indiquées par les segments épais. L'arrangement représenté à gauche détecte principalement le contour des tempes, alors que celui représenté à droite utilise les bords au dessus des yeux et sur la bouche. L'arrangement de gauche se décompose en deux sous arrangements de tailles trois, un de chaque coté du visage, chacun se décomposant à son tour en deux arrangements de tailles un et deux respectivement. Ces arrangements sont représentatifs des milliers que nous construisons. En fait, il apparaît que tous les arrangements sont composés de bords situés autour des yeux, de la bouche, du nez et sur le contours du visage. Cette accumulation est illustrée par la figure 6 sur laquelle l'intensité du noir de chaque pixel est proportionnel au nombre d'arrangements qui en dépendent.

Pour mettre en évidence l'influence de la complexité des arrangements, nous constituons un ensemble de test comprenant les 100 images de visages de la base ORL que nous n'avons pas utilisées lors de l'apprentissage, ainsi que 4000 sous-images ne comprenant pas de visages, prises dans des images diverse (scènes de rues, arbres, etc. Ces images ont été choisies pour être

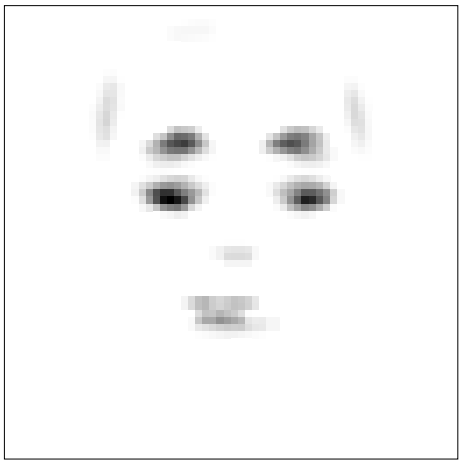

FIG. 6 – *Parties du visages utilisées par les arrangements. Les zones sombres sont les plus utilisées.*

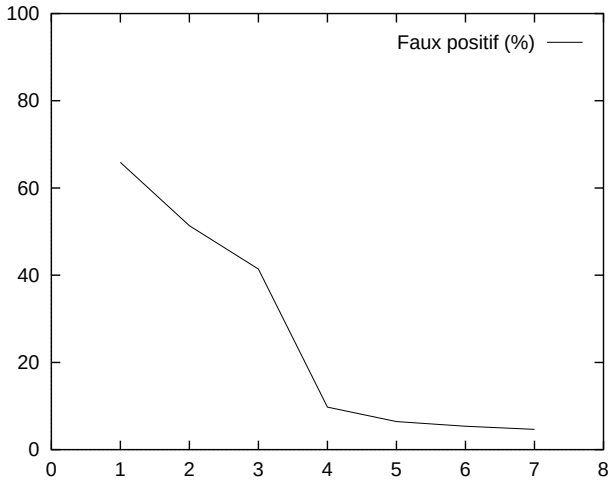

FIG. 7 – *Taux d'erreurs de faux-positifs fonction de la complexité des arrangements.*

riches en bords). Les visages, aussi bien à l'apprentissage qu'au test, sont recalés de manière à positionner les yeux et la bouche sur une pose standard.

Grâce au choix pessimiste des seuils $c_1, \ldots, c_n$, le nombre de faux négatifs peut être en pratique considéré comme nul. Le taux d'erreur de faux positif, exprimé en fonction de la complexité $\beta_k = \hat{P}_0(Z_1 \geq c_1, \ldots, Z_k \geq c_k)$ est représenté sur la figure 7. Le comportement important est la forte décroissance lorsque k augmente; la valeur finale (proche de 4%) est peu significative dans la mesure où elle ne correspond qu'à une seule étape du processus complet de détection hiérarchique. Par exemple, si nous restreignons à une tolérance de 2×2, le taux d'erreur tombe à moins de 1%.

8 Détecteur final

Le détecteur final utilise une succession de détecteurs, dédiés à des sous-ensembles de poses différents, et construit avec des tests élémentaires différents. La structures de la version que nous avons testée et dont nous donnons les résultats ne réalise pas exactement un partitionnement dichotomique de l'espace des poses. A la place, il correspond plutôt à une succession de passes sur l'image qui éliminent progressivement les emplacements où est susceptible de se trouver un visage. Chacune de ces passes utilise un ou plusieurs détecteur dédiés à un sous-ensemble de poses, et ne considère que les emplacements conservés par les passes qui l'ont précédée.

On peut appliquer un détecteur à une position x d'une image en extrayant la sous-image 32×32 dont le coin haut-gauche est à cette position. La détection se fait en appliquant successivement les détecteurs les uns après les autres, éliminant ainsi progressivement tous les emplacements qui ne correspondent pas à une sous-image contenant un visage. La précision de la détection est de 2×2. Le processus débute en marquant toutes les positions de coordonnées paires de l'image de la scène. Le premier détecteur a une tolérance de 4×4 pour la position de la bouche et une tolérance complète pour les autres paramètres de la pose; il est construit sur l'ensemble d'apprentissage complet, donc chaque image est dupliquée 16 fois afin que la bouche apparaisse à n'importe quelle position dans un carré 4×4. On applique ce détecteur en tous les points de l'image qui ont des coordonnées multiples de 4 (cf. figure 8). Comme les détecteurs sont construits avec la contrainte d'avoir un taux de faux négatifs de 0%, cette première passe n'élimine aucun site qui porte un visage. Le second détecteur est similaire au premier, mais n'admet qu'une tolérance de 2×2 en translation. On l'applique donc à toutes les positions qui ont été conservées par la première passe.

La passe suivante utilise deux détecteurs dédiés à des distances différentes entre les yeux. On partage l'ensemble d'apprentissage en deux moitié de même tailles, la première contenant les images de visages avec les plus petites distances entre les yeux, la seconde les autres. Ainsi, on construit deux détecteurs dédiés à des poses différentes. On ne conserve que les emplacement où au moins l'un des deux détecte un visage. On applique enfin une passe utilisant quatre détecteurs chacun dédié à un une inclinaison différente du visage. Les détecteurs de ces 4 passes sont construits en utilisant des tests élémentaires "positifs". Tous ces détecteurs sont croissants (combinaisons de conjonctions et de disjonctions). Ainsi, si on rajoute des bords dans une image, cela ne peux que créer de nouvelles détec-

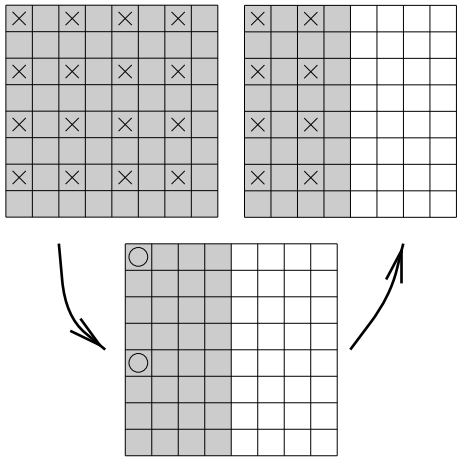

FIG. 8 – *Première étape de la détection finale. On commence par noter toutes les positions où peut se trouver un visages (on ne considère que les coordonnées paires, représentées par un $\times$). Là où $f^{4\times4}$ ne détecte pas de visage (absence de $\circ$), on peut éliminer 4 emplacements de la liste des emplacements de visages.*

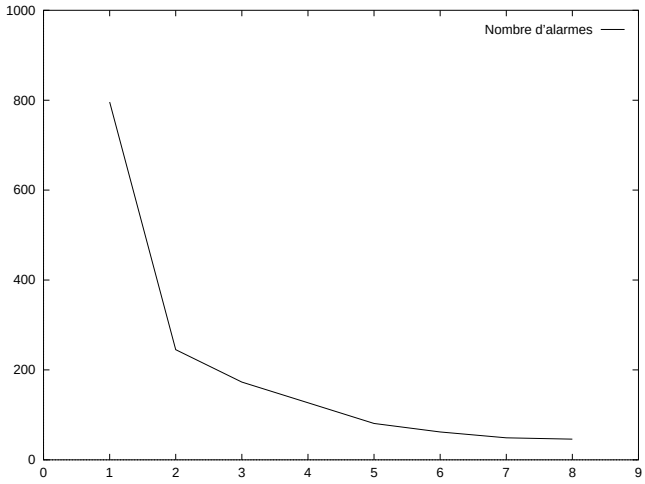

FIG. 9 – *Nombre de détections en fonction du nombre de passes effectuées. Les quatre premières passes utilisent des détecteurs construits à partir des tests élémentaires positifs, les quatre suivantes utilisent des détecteurs construits à partir des tests élémentaires négatifs.*

tions. En particulier une image très riche en bords (par exemple une texture avec de hautes fréquences) créera un grand nombre de faux positifs.
Pour éliminer des faux positifs créés dans de telles situations, on applique exactement le même processus que précédemment (c'est à dire la constructions des détecteurs pour les quatre passes successives), en utilisant comme tests élémentaires les tests négatifs, qui détectent l'absence de bords. Nous appliquons donc 4 passes supplémentaires, semblables aux 4 premières, mais basées sur des tests élémentaires négatifs. La figure 9 représente le nombre total de détections en fonction du nombre de passes effectuées.

9 Détection dans des scènes complètes

Nous avons testé le détecteur complet, à l'aide de plusieurs scènes[1]. Ces résultats sont préliminaires, mais très encourageants considérant la taille réduite de la base d'apprentissage. La scène représentée sur la figure 10 est représentative du type de résultats que nous obtenons. La tolérance du détecteur pour les différents degrés de liberté est liée aux variations de ces mêmes degrés de liberté dans la base d'apprentissage. Ainsi, sous la forme décrite ici, le détecteur tolère une variation de l'échelle de $\pm 25\%$ et une variation angulaire de $\pm 10°$. On peut augmenter cette tolérance en générant artificiellement des images dans la base de données.

1. Ces images sont issues de l'ensemble "C" d'image collectées à CMU par Henry A. Rowley, Shumeet Baluja, et Takeo Kanade.

De plus, pour pouvoir détecter des visages à toutes les échelles, on peut opérer la détection sur l'image en la sous-échantillonnant.

10 Conclusion

Nous avons formulé la détection comme un problème d'apprentissage dont les contraintes sont l'invariance et l'efficacité. Nous considérons qu'il est possible de faire cette détection en recherchant des structures qui sont beaucoup plus probables pour la loi des images d'objets que pour celle des images de fond. Nous gérons de manière *explicite* les contraintes naturelle - invariance à la photométrie, déformations géométriques, très faible nombre de faux négatifs et petite bases de données d'apprentissages.

L'apprentissage que nous proposons est très spécialisé pour la vision. Nous retrouvons la dépendance spatiale en estimant des corrélations entre des présences de structures localisées. Chacune de ces estimations est explicite, ce qui rend l'apprentissage très rapide.

Finalement, la structure globale du détecteur que nous avons développé est hautement hiérarchisée aussi bien dans l'espace des poses que dans la description des images relatives à une pose. Les résultats encourageants obtenus à l'aide de cette version préliminaire de l'algorithme devraient être améliorés par l'utilisation d'un partitionnement réellement hiérarchique de l'espace des poses.

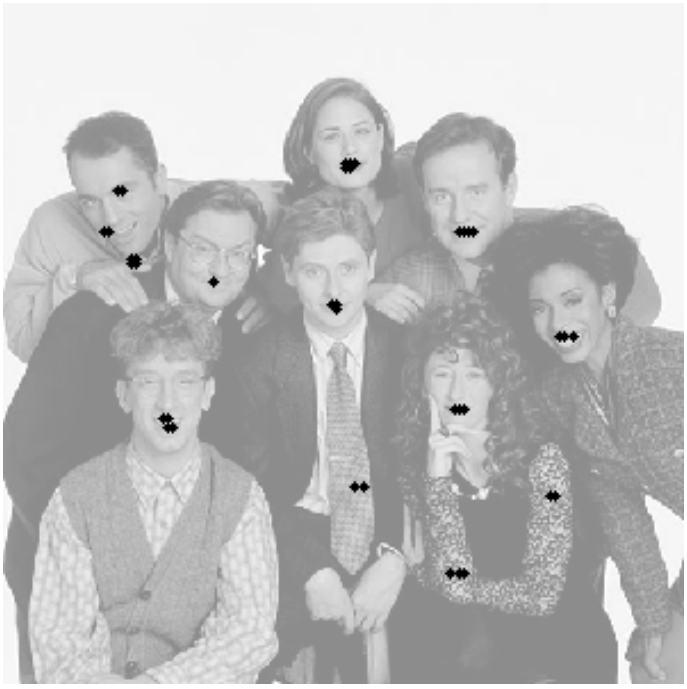

FIG. 10 – *Exemple de détection de visages sur une scène complète*

Références

[1] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9:1545–1588, 1997.

[2] Y. Amit and D. Geman. A computational model for visual selection. *Neural Computation*, 1999. To appear.

[3] T. F. Cootes and C. J. Taylor. Locating faces using statistical feature detectors. In *Proceedings, Second International Conference on Automatic Face and Gesture Recognition*, pages 204–209. IEEE Computer Society Press, 1996.

[4] D. Geman and B. Jedynak. An active testing model for tracking roads from satellite images. *IEEE Trans. PAMI*, 18:1–15, 1996.

[5] B. Jedynak and F. Fleuret. Reconnaissance d'objets 3d à l'aide d'arbres de classification. In *Proc. Image'Com 96*, Bordeaux, France, 1996.

[6] T. Leung, M. Burl, and P Perona. Finding faces in cluttered scenes using labeled random graph matching. In *Proceedings, 5th Int. Conf. on Comp. Vision*, pages 637–644, 1995.

[7] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *Proceedings, CVPR*, pages 130–136. IEEE Computer Society Press, 1997.

[8] H. A. Rowley, S. Baluja, and K. Takeo. Neural network-based face detection. *IEEE Trans. PAMI*, 20:23–38, 1998.

[9] K. K. Sung and T. Poggio. Example-based learning for view-based face detection. *IEEE Trans. PAMI*, 20:39–51, 1998.

[10] S. Ullman. *High-Level Vision*. M.I.T. Press, Cambridge, MA., 1996.

[11] K. Wilder. *Decision tree algorithms for handwritten digit recognition*. PhD thesis, University of Massachusetts, Amherst, Massachusetts, 1998.

[12] A. L. Yuille, D. S. Cohen, and P. Halliman. Feature extraction from faces using deformable templates. *Inter. J. Comp. Vision*, 8:104–109, 1992.