Modèles Génératifs et Efficacité Algorithmique pour la Prédiction

François Fleuret

11 Décembre 2006

Table des matières

Chapitre 1

Introduction

L'utilisation de techniques statistiques, et plus particulièrement de techniques d'apprentissage, s'est généralisée ces dernières années à un grand nombre de domaines de l'informatique appliquée. La vision artificielle a été particulièrement changée par une évolution de techniques exactes issues de la géométrie et de l'analyse vers des méthodes statistiques.

Cette évolution a induit le développement de techniques statistiques nouvelles capables de manipuler des signaux de très grandes dimensions, et débouchant sur des algorithmes utilisables en pratique. Les travaux résumés dans cette synthèse concernent le développement de nouvelles méthodes ayant de telles qualités.

Ce document se concentre sur deux principes essentiels sous-jacents aux travaux centraux de ma recherche, et les illustre par des exemples concrets. Il ne constitue pas une description exhaustive de mon travail et passe sous silence des voix de recherche latérales que j'ai développées sur la même période.

Le premier des principes illustrés dans cette synthèse est l'utilisation de modèles génératifs. La majeur partie des applications auxquelles j'ai été confronté peuvent être formalisées à l'aide d'un état caché aléatoire qui est la grandeur d'intérêt, et d'un signal visible, souvent de très grande dimension (une image, un vecteur de mesures). Il est souvent beaucoup plus facile de formaliser une connaissance *a priori* du problème à l'aide d'un modèle génératif qui spécifie la distribution de probabilité du signal étant donné l'état caché.

Un outil fondamental pour la conception de tels modèles génératifs consiste à introduire des grandeurs supplémentaires dans le modèle pour pouvoir légitimement représenter la distribution du signal de grande dimension conditionnellement à ces nouvelles variables comme une loi produit. Je présenterai dans le chapitre 2 en quoi ce principe a permis de résoudre efficacement des problèmes de suivi de personnes avec plusieurs caméras (Fleuret et al. 2005, Berclaz et al. 2006, Fleuret et al. 2006), de détection de visages rapide (Fleuret 2000, Fleu-

ret & Geman 2001, Fleuret & Geman 2002), de prédiction à partir de résultats d'amplification de gènes (Fleuret & Gerstner 2005) et d'apprentissage à partir d'un seul exemple (Fleuret & Blanchard 2005).

Le second principe sous-jacent aux travaux présentés ici est le contrôle du coût algorithmique, c'est à dire la prise en compte de la dualité entre les modèles mathématiques et leurs pendants algorithmiques. La plupart des problèmes réels pourraient être résolus de manière presque optimale avec des techniques naïves si l'on disposait d'une puissance de calcul infinie. Le développement d'une technique d'apprentissage statistique est donc toujours fait sous des contraintes computationnelles.

Je décrirai dans le chapitre 3 des travaux de recherche pour lesquels nous avons explicitement tenu compte de l'organisation algorithmique. Ces travaux regroupent une technique hiérarchique pour la détection de visages (Fleuret 2000, Fleuret & Geman 2001, Fleuret & Geman 2002), un algorithme de sélection de paramètres paresseux (Fleuret 2004) et des techniques de pré-calcul pour le suivi de personnes (Fleuret et al. 2005, Berclaz et al. 2006, Fleuret et al. 2006). Toutes ces méthodes, qu'elles soient purement algorithmiques ou qu'elles reposent sur des approximations et des stratégies adaptatives, permettent finalement de manipuler des modèles mathématiques extrêmement puissants tout en gardant une vitesse de traitement utilisable en pratique.

Chapitre 2

Modèles génératifs

Les techniques d'apprentissage statistique se partagent en deux familles disjointes : d'une part les techniques "discriminatives", et d'autre part les techniques "génératives" (Ng & Jordan 2002, Ulusoy & Bishop 2005). Pour résumer en quelques mots, on peut dire que les premières se limitent à trouver des critères pour faire de la prédiction à partir de signaux sans jamais préciser la structure de ces derniers, alors que les secondes reposent sur une modélisation du signal, pour éventuellement réaliser une tâche de prédiction.

Un exemple simple permet de préciser les différences entre les deux approches. Considérons la prédiction du sexe d'un individu (état caché S, variable aléatoire à valeur dans $\{m,\, f\}$) en fonction de sa taille (signal visible T variable aléatoire réelle).

Une démarche discriminative consisterait à définir un ensemble de fonctions de classification, par exemple ici des seuillages $f_\alpha(x) = 1_{x \geq \alpha}$, puis à en choisir une lors de l'apprentissage pour minimiser une fonction d'erreur empirique.

Une approche générative en revanche modéliserait le signal conditionnellement à l'état caché $\mu(T \,|\, S = s)$ pour ensuite en déduire la règle de prédiction. Le modèle serait donc introduit via une famille de densités μ_α, et l'apprentissage consisterait à choisir les paramètres α des densités concernées. La prédiction pourrait être le calcul d'un maximum *a posteriori* $s^\star = \arg\max_s P(S = s \,|\, T = t)$ ou l'estimation de la probabilité postérieure elle-même $P(S = s \,|\, T = t)$.

Le choix de méthode paraît ici évident. Si l'on a simplement besoin d'un résultat "dur" de classification ou de prédiction, les méthodes discriminatives semblent n'avoir que des avantages. Leur intérêt dans un tel cas est double. D'une part elles évitent d'investir de la ressource dans la modélisation de structures qui n'ont pas d'intérêt pour le problème que l'on considère. Dans notre exemple, cela se traduit par l'inutilité de modéliser finement la distribution des tailles des femmes petites ou des hommes grands. Et d'autre part elles effectuent implicitement le

calcul du maximum *a posteriori* dans le cas génératif, qui et souvent coûteux algorithmiquement.

En pratique la situation est nettement plus nuancée. Lorsque l'on est confronté à un nouveau problème relevant de l'apprentissage statistique, il est beaucoup plus facile d'introduire des connaissances relatives au problème via une méthode générative, même si l'objectif est purement discriminatif. Cette facilité à introduire des connaissances dans le modèle réduit souvent le nombre d'exemples d'apprentissage nécessaires. De plus, la conception d'un schéma génératif clarifie les hypothèses d'indépendance et les choix de modèles de densités qui ont été faits pour modéliser le problème et permet d'expliciter les conditions de fonctionnement ou d'échec de l'algorithme résultant.

L'objet de ce premier chapitre est de présenter quatre problèmes différents qui ont été traités avec succès à l'aide de méthodes génératives.

2.1 Notion d'indépendance conditionnelle

Un principe fondamental qui revient souvent dans les travaux présentés dans ces pages est l'indépendance conditionnelle de variables visibles, étant données les valeurs de variables cachées. C'est une généralisation de l'idée sur laquelle repose le schéma *naïf Bayésien* de classification (Duda & Hart 1973, Langley et al. 1992) utilisé par exemple pour filtrer les mails indésirables.

Modéliser une loi jointe de manière explicite, même avec des modèles tels que des mixtures de gaussiennes, est illusoire pour des signaux de grandes dimensions tels que des images ou des échantillons sonores. De surcroît le faire de manière aveugle et générique est contre-intuitif puisque l'on sait que le signal est fortement structuré. L'indépendance conditionnelle offre un outil général pour construire un modèle de densité incorporant la connaissance *a priori* du problème. Il a été utilisé avec succés en particulier en vision (Amit 2002).

2.1.1 Un problème jouet

Considérons par exemple un problème jouet qui consiste à localiser un carré noir dans une image blanche, le signal étant bruité par un bruit qui induit une permutation des pixels avec une probabilité ϵ. Les inconnues (X, Y) sont deux variables aléatoires réelles indépendantes qui représentent la position du centre de la cible, et le signal visible est une carte de pixels binaires I, variable aléatoire à valeurs dans $\{0, 1\}^{L \times H}$ où L et H représentent la taille de l'image.

Le processus algorithmique qui produit l'image I génère indépendamment X et Y, puis tous les pixels de l'image, avec une probabilité ϵ d'être noir en dehors de

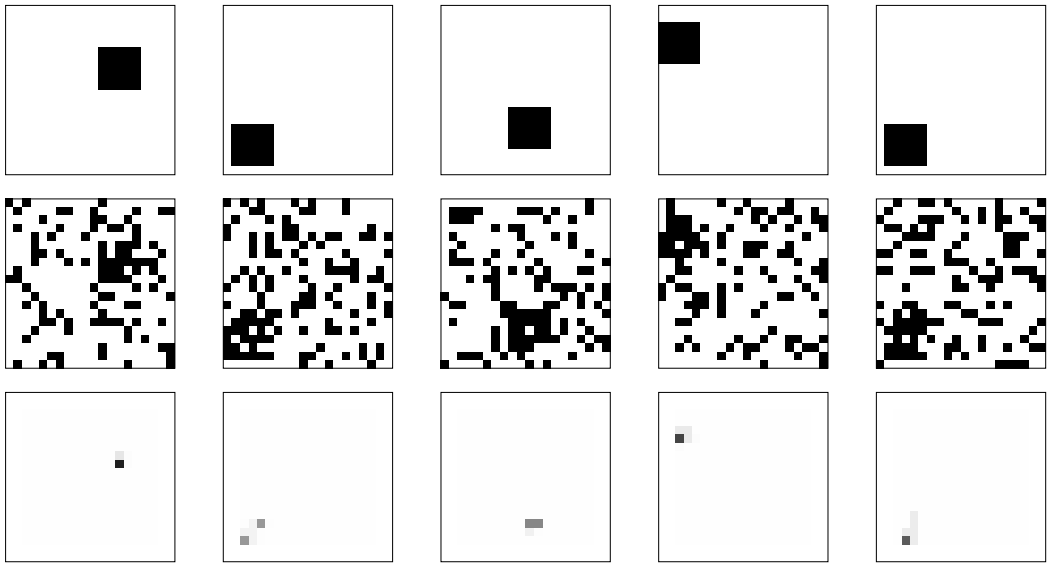

FIG. 2.1 – Chaque colonne montre une réalisation du modèle pour $L = H = 20$, $T = 5$ et $\epsilon = 0.3$. La première ligne montre la position de la cible sans le bruit, la deuxième ligne le signal I, et la troisième ligne la carte de scores $P(X = x, Y = y \mid I)$.

la cible carrée de taille T centrée sur (X, Y), et $1 - \epsilon$ à l'intérieur de cette cible.

Les variables X et Y sont donc indépendantes et uniformes, et conditionnellement à X et Y, les pixels $I(x, y)$ sont des variables de Bernoulli indépendantes, de paramètre $1-\epsilon$ pour $X-T/2 \leq x < X+(T+1)/2, Y-T/2 \leq y < Y+(T+1)/2$ et de paramètre ϵ sinon.

Intuitivement, il est évident que sans conditionnement, les pixels ne sont pas indépendants. Par exemple, connaître les valeurs de tous les pixels de l'image sauf un permet de prédire où se trouve la cible, et savoir où se trouve la cible permet de déterminer si ce pixel inconnu est plutôt noir ou blanc. En revanche, si l'on connaît déjà X et Y, avoir accès aux valeurs de certains pixels ne donne aucune information nouvelle sur un autre pixel.

2.1.2 Règle de détection

De ce modèle on obtient, en notant $\xi(\epsilon, 0) = \log(1-\epsilon)$, $\xi(\epsilon, 1) = \log \epsilon$ et $A(x, y) = \{(\alpha, \beta) : x - T/2 \leq \alpha < x + (T + 1)/2, y - T/2 \leq \beta < y + (T + 1)/2\}$

$$
\begin{aligned}
&\log P(X = x,\, Y = y \,|\, I) \\
&= \log \frac{P(I \,|\, X = x,\, Y = y)\, P(X = x,\, Y = y)}{P(I)} \\
&= \log P(I \,|\, X = x,\, Y = y) + \log \frac{P(X = x, Y = y)}{P(I)} \\
&= \sum_{\alpha,\beta} \log P(I(\alpha,\beta) \,|\, X = x,\, Y = y) + \log \frac{P(X = x, Y = y)}{P(I)} \\
&= \sum_{\alpha,\beta \in A(x,y)} \xi(I(\alpha,\beta), 1-\epsilon) + \sum_{\alpha,\beta \notin A(x,y)} \xi(I(\alpha,\beta), \epsilon) + \log \frac{P(X = x, Y = y)}{P(I)} \\
&= \sum_{\alpha,\beta \in A(x,y)} \xi(I(\alpha,\beta), 1-\epsilon) - \xi(I(\alpha,\beta), \epsilon) + \sum_{\alpha,\beta} \xi(I(\alpha,\beta), \epsilon) + \log \frac{P(X = x, Y = y)}{P(I)} \\
&= \log \left(\frac{1-\epsilon}{\epsilon}\right)^2 \sum_{\alpha,\beta \in A(x,y)} I(\alpha,\beta) + \zeta
\end{aligned}
$$

Où le terme ζ est indépendant de x et y et est le logarithme d'un terme de normalisation pour que la somme des $P(X = x, Y = y \,|\, I)$ soit égale à 1. Ce terme peut être ignoré et la normalisation obtenue de manière numérique.

Ainsi, de ce modèle nous déduisons une expression simple pour la probabilité conditionnelle de X, Y étant donné le signal I. De surcroît, cette expression est très intuitive, puisque son logarithme prend la forme d'un sommage sur les pixels de la cible à la position testée et est d'autant plus élevée qu'il y a de pixels noirs.

Cet exemple jouet illustre parfaitement la puissance d'une telle approche.

Non seulement nous avons obtenu une règle simple et très intuitive, mais nous avons en plus précisé très exactement les hypothèses sous lesquelles ce schéma est valide, et comment le modèle peut être modifié si une de ces hypothèses ne l'est plus. Si par exemple la loi *a priori* sur X et Y n'est plus uniforme, nous verrons naturellement apparaître un terme correctif correspondant au log de cette probabilité *a priori*. S'il existe des structures de corrélation entre pixels, il faudra simplement rajouter des variables supplémentaires pour obtenir l'indépendance, et les marginaliser.

Enfin, la principale faiblesse de cette approche apparaît dans le coût algorithmique de la procédure nécessaire pour calculer le maximum *a posteriori*. Sans hypothèse supplémentaire, et sans astuce algorithmique ou approximation, nous aurions dans ce cas précis un coût linéaire avec le nombre de positions (X, Y) à examiner et avec la surface de la cible.

FIG. 2.2 – Résultats de détection de visages frontaux à l'aide de la méthode Coarse-to-Fine.

2.2 Détection de visages

La détection de visages frontaux a constitué un des thèmes majeurs d'application de l'apprentissage statistique en vision au début des années 2000. En plus de nos propres travaux, plusieurs algorithmes ont depuis démontré que cela pouvait être effectué avec un taux d'erreur acceptable pour des applications réelles (Rowley et al. 1998, Viola & Jones 2001).

Mon travail de thèse, effectué sous la direction de Donald Geman, a consisté à développer un nouvel algorithme de détection rapide de visages (Fleuret & Geman 1999, Fleuret & Geman 2000, Fleuret & Geman 2001, Fleuret & Geman 2002).

Ce travail repose sur un modèle génératif de densité des images de visages conditionnellement à un état caché de *pose* qui incorpore les informations géométriques sur le positionnement du visage dans le plan image. À partir de ce modèle nous avons conçu une procédure adaptative de tests séquentiels qui demande un petit nombre d'exemples d'apprentissage et est très efficace algorithmiquement. Ce second aspect d'efficacité sera développé en détail au chapitre 3 (section 3.1, page 25).

2.2.1 Détection à l'aide d'un classifieur

L'objet d'un détecteur de visages est de traiter une *scène* qui est une image où apparaissent des visages vus de face, et de produire en résultat une liste de visages, chacun localisé dans l'image (cf. figure 2.2). La plupart des techniques

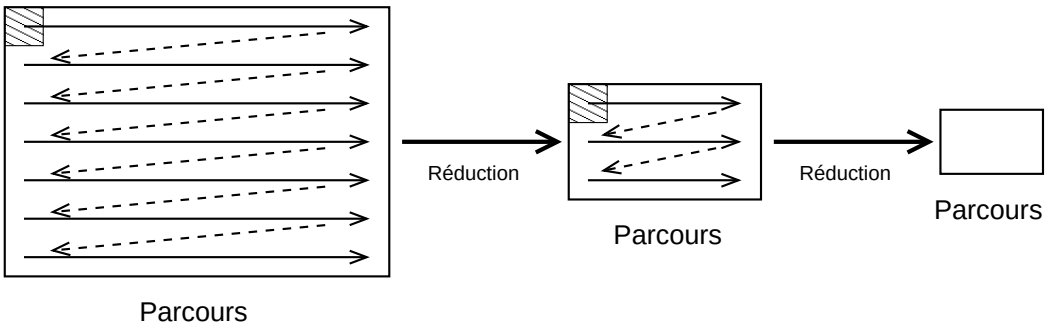

FIG. 2.3 – Le processus de détection consiste à parcourir toutes les positions de l'image à traiter à différentes échelles.

FIG. 2.4 – Nous définissons la position d'un visage dans le plan image à l'aide de quatre variables cachées : la position du centre des yeux X, Y, la distance entre les yeux S et l'inclinaison dans le plan image Θ.

existantes reposent sur un parcours de toutes les positions de l'image, à toutes les échelles possibles (cf. figure 2.3), combiné avec un classifieur à deux classes qui considère la sous-image correspondante et produit en retour un résultat booléen sur la présence ou l'absence d'un visage.

Finalement, la tâche centrale est donc effectuée par un classifieur à deux classes

$$f : \mathcal{I} \rightarrow \{0, 1\} \tag{2.1}$$

où $\mathcal{I} = [0, 1]^{64\times64}$ est l'espace des imagettes 64×64.

2.2.2 Somme de présences de bords

Dans le modèle d'apparence des visages que nous avons proposé, nous considérons un imagette $I \in [0, 1]^{64\times64}$ de taille 64×64 pixels, et nous introduisons un

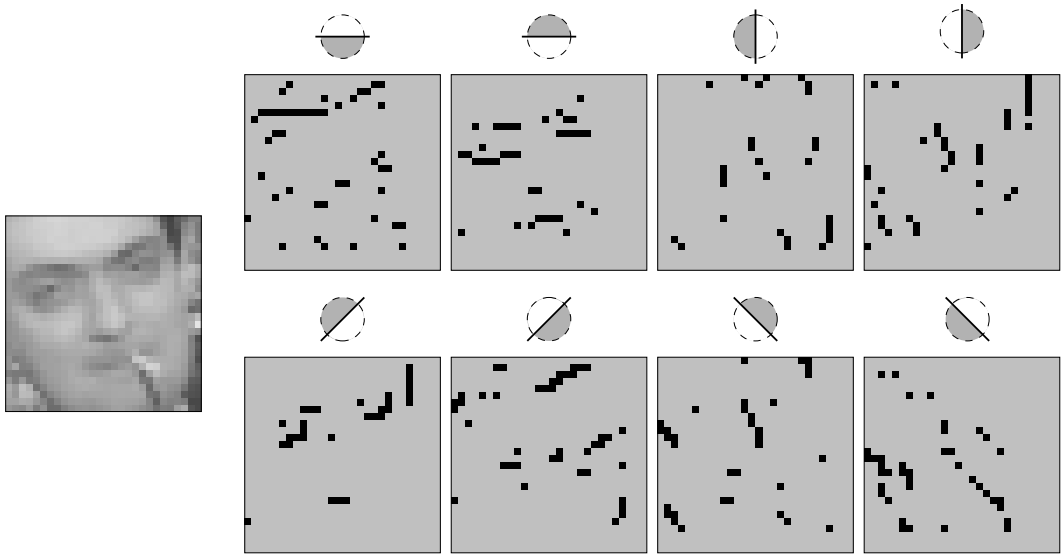

FIG. 2.5 – Les imagettes considérées sont initialement recodées à l'aide de détecteurs de bords.

état caché Γ composé de quatre paramètres de pose : la position du centre des yeux (X, Y) pour la localisation en translation, la distance entre les yeux S pour l'échelle, et l'inclinaison Θ (cf. figure 2.4).

Une imagette I est initialement recodée à l'aide d'un simple détecteur de bords qui produit en chaque pixel x, y une liste de 8×8 variables booléennes $E(x, y, \theta, t)$, chacune représentant la présence d'un bord à une orientation donnée θ, avec une tolérance en position donnée t.

La figure 2.5 montre la détection de ces bords avec une tolérance d'un seul pixel.

Nous faisons une hypothèse d'indépendance conditionnelle des variables $E(x, y, \theta, t)$ étant donné la pose Γ. En notant Y la classe de l'imagette (0 si elle ne contient pas de visage, et 1 sinon) et Γ la pose,

$$\exists (x_1, y_1, \theta_1, t_1), \ldots, (x_N, y_N, \theta_N, t_N),$$
$$P(E(x, y, \theta, t) \,|\, Y = 1, \Gamma) = \prod_n P(E(x_n, y_n, \theta_n, t_n) \,|\, Y = 1, \Gamma)$$

alors un test du type $P(E(x, y, \theta, t) \,|\, Y = 1, \Gamma) \geq e^T$ prend la forme

$$\sum_n \left\{ \log \frac{P(E(x_n, y_n, \theta_n, t_n) = 1 \,|\, Y = 1, \Gamma)}{P(E(x_n, y_n, \theta_n, t_n) = 0 \,|\, Y = 1, \Gamma)} \right\} E(x_n, y_n, \theta_n, t_n) \geq T \qquad (2.2)$$

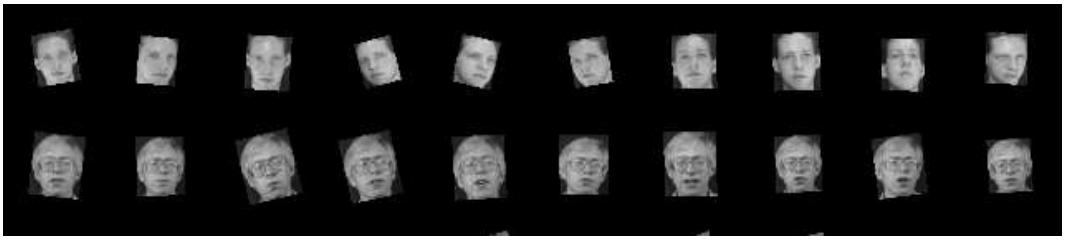

FIG. 2.6 – Exemple positif d'apprentissage. La pose Γ de chaque exemple est forcée à une certaine valeur en appliquant une translation, une rotation et un facteur d'échelle à l'image originale.

Dans la version la plus récente de cet algorithme (Fleuret & Geman 2002), nous sélectionnons à l'aide d'une procédure proche du boosting une sous-famille $(x_1, y_1, \theta_1, t_1), \ldots, (x_N, y_N, \theta_N, t_N)$ pour construire un classifieur binaire de la forme suivante

$$\sum_n E(x_n, y_n, \theta_n, t_n) \geq T \tag{2.3}$$

avec un seuil T maximum. Cette règle est donc construite à l'aide d'une approche discriminative, mais elle est justifiée par un modèle simple d'indépendance des variables $E(x, y, \theta, t)$ conditionnellement à la pose Γ. Nous verrons en §3.1 comment un grand nombre de tels classifieurs dédiés à des poses particulières sont ensuite combinés pour concevoir le détecteur complet.

2.3 Suivi de personnes multi-caméras

La détection de personnes à l'aide d'un système vidéo multi-caméras consiste à traiter en entrée plusieurs flux vidéo correspondant à la même scène filmée depuis plusieurs angles de vues différents, et à produire en sortie une information sur la présence d'individus dans la scène comme illustré sur la figure 2.7.

Dans ce projet (Fleuret et al. 2005, Berclaz et al. 2006, Fleuret et al. 2006) nous avons proposé de modéliser le signal produit par les caméras étant donné un état caché de présence de personnes. De ce modèle nous avons déduit une expression analytique de la probabilité d'occupation du sol étant donné le signal, expression dont nous dérivons un schéma itératif d'estimation.

FIG. 2.7 – Exemples de résultats de suivi de personnes avec un système à quatre caméras. Les boîtes montrent le résultat de l'algorithme complet qui optimise les trajectoires sur plusieurs dizaines de frames selon un modèle combinant l'apparence et le mouvement.

2.3.1 Modélisation

Pour des raisons applicatives, nous avons dû initialement traiter des données produites par un algorithme de segmentation reposant sur le mouvement. Un exemple de résultat de cette segmentation est représenté sur la ligne du bas de la figure 2.8. Nous avions alors accès uniquement à ce signal binaire, et à une seule frame temporelle.

Pour modéliser le problème, nous discrétisons le sol de la zone d'intérêt en un nombre fini de positions avec une résolution d'une vingtaine de centimètres. À chacune de ces positions n, nous associons une variable aléatoire booléenne X_n qui représente la présence d'une personne à cet endroit. Si nous notons $B^1, \ldots, B^C$ des variables aléatoires à valeur dans $\{0,1\}^{L\times H}$ représentant les images binaires produites par la segmentation basée sur le mouvement, nous voudrions idéalement donner une estimation de

$$P(X_1, \ldots, X_N \,|\, B^1, \ldots, B^C) \tag{2.4}$$

c'est à dire de la loi jointe conditionnellement aux images.

L'algorithme original que nous avons proposé repose sur un modèle de l'image étant donnée l'état caché et sur une approximation de la loi jointe conditionnelle (2.4) par une loi produit. Les marginales de cette dernière sont obtenues en minimisant la divergence de Kulback-Leibler avec la vraie loi postérieure. Nous entendons ici par "vraie loi postérieure" celle qui correspond à un modèle légitime de $P(B^1, \ldots, B^C \,|\, X_1, \ldots, X_N)$ et à un modèle produit de la loi *a priori* $P(X_1, \ldots, X_N)$.

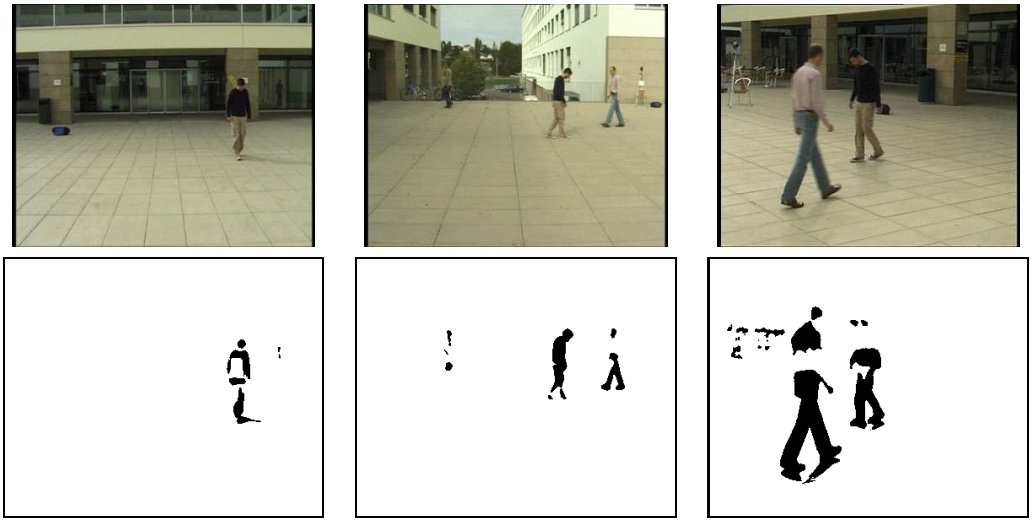

FIG. 2.8 – Résultats de la segmentation basée sur le mouvement.

2.3.2 Modèle de synthèse grossier

Le modèle d'apparence $P(B^1, \ldots, B^C \mid X_1, \ldots, X_N)$ que nous avons utilisé repose sur une hypothèse d'indépendance conditionnelle des différentes vues étant donné l'état caché et sur un modèle génératif de chaque vue. Ce dernier constitue un point essentiel de notre approche.

Nous introduisons un processus grossier de synthèse d'image. En utilisant la calibration géométrique des caméras, nous associons à chaque position n et chaque caméra c un rectangle $\mathcal{A}_n^c$ correspondant à une silhouette humaine de 175cm de haut et 30cm de large située en n et vue depuis c. Nous notons A^c une image fonction de $X_1, \ldots, X_N$ obtenue en plaçant un tel rectangle $\mathcal{A}_n^c$ en toute position où $X_n = 1$, tel que c'est illustré sur la figure 2.9.

Nous modélisons alors le signal $B^1, \ldots, B^C$ étant donné l'état caché $X_1, \ldots, X_N$ par

$$
\begin{aligned}
P(B^1, \ldots, B^C \mid X_1, \ldots, X_n) &= \prod_c P(B^c \mid X_1, \ldots, X_n) && (2.5)\\
&= \prod_c P(B^c \mid A^c) && (2.6)\\
&= \frac{1}{Z} \prod_c \exp\left(-\Psi(B^c, A^c)\right) && (2.7)
\end{aligned}
$$

où $\Psi(B^c, A^c) = \frac{\|B^c - A^c\|_1}{\sigma \|A^c\|_1}$ et Z est un facteur de normalisation.

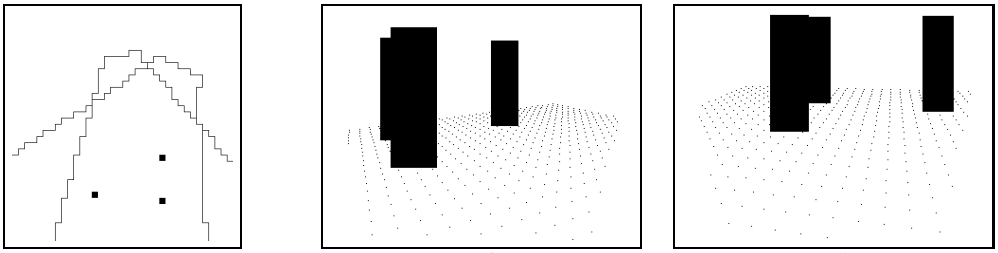

FIG. 2.9 – Nous introduisons un modèle de synthèse grossier de l'image étant donné l'état caché $X_1, \ldots, X_N$.

2.3.3 Approximation à l'aide d'une loi produit

Même sous ce modèle, la loi jointe postérieure $P(X_1, \ldots, X_N \,|\, B^1, \ldots, B^C)$ est trop complexe pour être manipulée ou estimée. De plus, la plupart des applications utilisant du suivi multi-caméras nécessitent simplement une estimation des probabilités marginales $P(X_n \,|\, B^1, \ldots, B^C)$.

Le schéma que nous proposons consiste à introduire une loi produit $Q(X_1, \ldots, X_N)$ et à estimer les probabilités marginales $q_n = Q(X_n = 1)$ de façon à minimiser la divergence de Kullback-Liebler entre Q et la "vraie" loi postérieure $P(\cdot \,|\, B^1, \ldots, B^C)$ sous notre modèle. Si nous notons ϵ_n la probabilité marginale *a priori* $P(X_n = 1)$, $\lambda_n = \log \frac{\epsilon_n}{1-\epsilon_n}$ et E_Q l'espérance pour $X_1, \ldots, X_N \sim Q$, alors

$$\begin{aligned}
&\frac{\partial}{\partial q_n} KL(Q,\, P(\,\cdot\,|\,\mathbf{B})) \\
&= E_Q\left(\sum_c \Psi(B^c, A^c) \;\middle|\; X_n = 1\right) - E_Q\left(\sum_c \Psi(B^c, A^c) \;\middle|\; X_n = 0\right) \\
&\quad + \log \frac{q_n\,(1-\epsilon_n)}{(1-q_n)\,\epsilon_n}
\end{aligned} \tag{2.8}$$

ainsi, résoudre

$$\frac{\partial}{\partial q_n} KL(Q,\, P(\,\cdot\,|\,\mathbf{B})) = 0 \tag{2.9}$$

amène à

$$q_n = \frac{1}{1 + \exp\left(\lambda_n + \sum_c E_Q\left(\,\Psi(B^c, A^c) \,|\, X_n = 1\right) - E_Q\left(\,\Psi(B^c, A^c) \,|\, X_n = 0\right)\right)}.$$

Nous avons donc une expression des q_n sous la forme d'un large système d'équations. Nous verrons au chapitre 3 (section 3.3, page 30) comment ce système peut être approché puis résolu numériquement de manière à traiter des séquences vidéo à raison de plusieurs images par seconde.

2.4 Prédiction de caractères phénotypiques

Avec le développement de techniques d'amplifications de gènes, la prédiction à partir d'informations génétiques est devenue un des objectifs prioritaires des techniques d'apprentissage.

Le problème auquel nous nous sommes intéressés dans cette étude (Fleuret & Gerstner 2005) consiste à prédire des propriétés électro-physiologiques de neurones à partir du résultat d'une procédure de RT-PCR (*reverse transcriptase polymerase chain reaction*) qui estime des niveaux d'expressions de gènes dans une cellule. Cette procédure est connue pour souffrir d'un fort taux de faux négatifs, c'est à dire de défauts d'amplification de certains gènes exprimés, et d'un faible taux de faux positifs.

Nous avons proposé un modèle génératif du résultat de RT-PCR étant donné les véritables valeurs d'expressions des gènes dans la cellule. De ce modèle nous en avons déduit une mesure de similarité entre cellules qui prend en compte l'asymétrie des erreurs. Ainsi, deux cellules qui se ressemblent parce qu'elles ont des gènes amplifiés en commun sont plus similaires que deux cellules qui se ressemblent parce qu'elles ont des gènes non-amplifiés en commun, ces derniers pouvant ne pas avoir été amplifiés par erreur. Nous montrons que cette mesure de similarité est un noyau quasi-conforme qui vérifie donc la propriété de Mercer et peut être utilisé comme noyau pour des techniques telles que les SVMs (Vapnik 1998, Christiani & Shawe-Taylor 2000).

2.4.1 Asymétrie des erreurs de mesures

Nous disposions pour cette étude de mesures relatives à 200 neurones. Pour chacune de ces cellules, une procédure de *patch-clamp* (Hamill et al. 1981, Sakmann & Neher 1983), décrite sur la figure 2.10, était utilisée pour mesurer les propriétés électro-physiologiques de la cellule ainsi que pour extraire un échantillon à soumettre ultérieurement à la RT-PCR (Toledo-Rodriguez et al. 2004).

Au cours de cette dernière, les fragments d'ARN sont retranscrits en ADN par la transcriptase, puis les fragments d'ADN obtenus sont amplifiés exponentiellement par la polymérase. Finalement, les fragments d'ADN ainsi dupliqués sont placés dans un gel et soumis à un champ électrique qui les sépare et permet de les

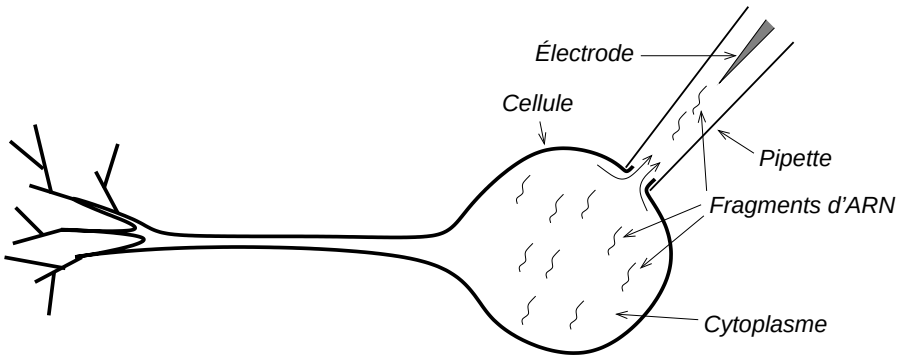

FIG. 2.10 – Procédure de *patch-clamp*. La membrane de la cellule est cassée par succion. Une partie du cytoplasme remonte dans la pipette et entre en contact avec une électrode, ce qui permet d'effectuer des mesures électro-physiologiques. Des fragments d'ARN sont également aspirés dans la pipette et peuvent être soumis plus tard à la procédure de RT-PCR.

identifier.

La véritable information d'intérêt pour les biologistes est de savoir quels sont les gènes exprimés dans la cellule. Ces derniers induisent la présence des fragments d'ARN, qui à leur tour induisent la mesure lors de la procédure de RT-PCR.

La détection d'ADN étranger est facile à contrôler en utilisant des procédures qui évitent la contamination, alors qu'il arrive fréquemment pour des raisons purement physiques qu'aucun fragment d'ARN relatif à un gène exprimé ne soit pas capturé lors du *patch-clamp*, ou que le fragment ne soit pas retranscrit, ou enfin que l'ADN ne soit pas amplifié.

L'objet de ce travail était de trouver un moyen d'exploiter cette asymétrie. Nous avons montré que l'on peut concevoir une mesure de similarité qui a les propriétés d'un noyau, et qui dérive d'un modèle légitime du bruit lors de la mesure.

2.4.2 Modèle de la procédure de RT-PCR

Étant donné le nombre N de gènes mesurés lors de la procédure de RT-PCR, nous notons $\mathbf{X} = (X^1, \ldots, X^N)$ un vecteur aléatoire de $\{0, 1\}^N$ qui représente le résultat de ladite procédure. L'amplification exponentielle étant très instable, ignorer le résultat quantitatif et se retreindre à une réponse dans $\{0, 1\}$ n'est pas un approximation coûteuse. Nous notons $\mathbf{Z} = (Z^1, \ldots, Z^N)$ un vecteur aléatoire de $\{0, 1\}^N$ qui représente la "véritable" expression des gènes. Cet état est inconnu.

Nous proposons le modèle génératif suivant :

$$P(X^1, \ldots, X^N \mid Z^1, \ldots, Z^N) \;=\; \prod_n P(X^n \mid Z^n) \tag{2.10}$$

avec

$$\begin{aligned} P(X^n = 0 \mid Z^n = 0) &= 1 \\ P(X^n = 0 \mid Z^n = 1) &= \epsilon \\ P(X^n = 1 \mid Z^n = 0) &= 0 \\ P(X^n = 1 \mid Z^n = 1) &= 1 - \epsilon \end{aligned}$$

où ϵ est le taux de faux négatifs, que nous fixons à 0.25, ce qui correspond aux estimations fournies par des biologistes.

2.4.3 Mesure de similarité comme noyau

Nous avons proposé comme mesure de similarité entre deux cellules la probabilité que les vecteurs de gènes exprimés soient les mêmes, conditionnellement aux mesures :

$$k(x_1,\, x_2) = P(\mathbf{Z}_1 = \mathbf{Z}_2 \mid \mathbf{X}_1 = x_1,\, \mathbf{X}_2 = x_2) \tag{2.11}$$

On peut montrer que sous notre modèle, on obtient alors

$$k(x_1,\, x_2) = \prod_{n=1}^{N} \kappa_n(x_1^n, x_2^n) \tag{2.12}$$

avec

$$\kappa_n(a, b) = \sum_{c \in \{0,1\}} P(Z_1^n = c \mid X_1^n = a)\; P(Z_2^n = c \mid X_2^n = b) \tag{2.13}$$

Enfin, comme ce noyau est défini sur des vecteur de $\{0,\, 1\}^N$, on peut le mettre sous la forme

$$k(x_1,\, x_2) \;\propto\; \exp\left(\delta ||x_1||^2 + \delta ||x_2||^2 + \gamma\, ||x_1 - x_2||^2 \right),$$

avec $\gamma < 0$ et $\delta > 0$.

On remarquera que le signe de δ serait opposé si les erreurs de faux positifs étaient plus fréquentes que les erreurs de faux négatifs.

Une telle expression est celle d'un noyau quasi-conforme (Smola & Schölkopf 1998, Amari & Wu 2000), qui a donc la propriété de Mercer et peut être utilisé avec des techniques classiques d'apprentissage à noyaux (Smola & Schölkopf 1998), telle que les SVMs (Vapnik 1998, Christiani & Shawe-Taylor 2000). Ce résultat justifie également l'utilisation d'un noyau Gaussien, sous hypothèse que les erreurs sont symétriques.

Nos expériences avec des données réelles (Fleuret & Gerstner 2005) ont démontré la cohérence de notre modèle avec l'hypothèse de fort taux de faux négatifs, et les bonnes performances d'une SVM utilisant cette mesure de similarité par comparaison avec des méthodes linéaires classiques.

2.5 Apprentissage à partir d'un exemple unique

Bien que la plupart des animaux soient capables d'apprendre l'apparence d'un objet à partir d'un très petit nombre d'exemples, les techniques d'apprentissage statistique nécessitent pour la plupart des ensembles d'apprentissage de grande taille. Nous nous sommes intéressés dans ce projet (Fleuret & Blanchard 2005) au cas limite où l'on ne dispose que d'une seule image d'un objet pour en apprendre l'apparence.

La procédure que nous avons développée comprend une première phase d'apprentissage durant laquelle nous avons accès à un grand nombre d'images d'un grand nombre d'objets. Au cours de cette phase, l'algorithme construit plusieurs dizaines de coupures de l'espace du signal à l'aide de perceptrons très invariants sur les images d'apprentissage. Précisément, ces perceptrons sont construits de manière à avoir une réponse constante sur toutes les images de n'importe quel objet. Cette réponse peut être 0 ou 1 de manière indifférente, mais elle doit être constante sur toutes ses images.

Lors de la phase de test, étant données deux images de nouveaux objets, c'est à dire d'objets qui n'ont pas été vus pendant l'apprentissage, nous utilisons comme critère de comparaison la probabilité postérieure que les deux objets soient de la même classe, étant données les réponses des perceptrons. Comme nous le verrons à la section §2.5.2 la règle exacte de comparaison découle d'un modèle génératif de la réponse des M perceptrons étant donnée une signature exacte dans $\{0, 1\}^M$ associée aux images, inconnue, mais parfaitement invariante.

2.5.1 Coupure invariante

Nous appelons coupure invariante une fonction F de l'espace du signal dans $\{0, 1\}$ qui a la propriété d'être fortement déterminée par la classe du signal, et malgré tout fortement aléatoire, c'est à dire, en notant C la vraie classe de l'image, donc l'identité de l'objet représenté, $H(F \mid C) \simeq 0$ et $H(F) \simeq 1$.

La première partie de notre approche consiste à construire un grand nombre de telles coupures. Chacune est obtenue en associant le label 0 aux images de la moitié des objets (choisis aléatoirement), et 1 aux images des autres objets, puis en entraînant un perceptron à prédire ce label.

Étant donné un ensemble d'apprentissage composé d'images, chacune accompagnée d'un "vrai" label booléen, nous commençons par calculer les réponses de $50,000$ détecteur de bords, puis nous en sélectionnons un sous-ensemble de $2,000$ en utilisant une technique reposant sur l'information mutuelle conditionnelle (Fleuret 2004). Enfin, nous entraînons un perceptron avec une procédure classique (Christiani & Shawe-Taylor 2000, page. 12–14).

Dans la suite, pour une image donnée, nous notons C, variable aléatoire sur $\mathbb{N}$, la vraie classe de l'image (c'est à dire l'index de l'objet qu'elle représente), $\mathbf{L} = (L_1, \ldots, L_M)$, variable aléatoire sur $\mathbb{R}^M$, les réponses des perceptrons avant seuillage et enfin $\mathbf{S} = (S_1, \ldots, S_M)$, variable aléatoire sur $\{0, 1\}^M$, les "vrais labels" de l'objet, labels que les perceptrons essayent de prédire. Seuls les $L_1, \ldots, L_M$ sont connus lors du test.

2.5.2 Combinaison des coupures

Étant données deux nouvelles images représentant des objets inconnus et jamais vus lors de l'apprentissage, notre objectif est de prédire si ces deux objets sont les mêmes. Nous notons avec un exposant 1 ou 2 les grandeurs relatives à chacun de ces deux images. Par exemple C^1 et C^2 sont les vraies classes de ces objets.

Intuitivement, il paraît logique d'utiliser comme critère de similarité pour comparer deux images le nombre de perceptrons qui répondent de manière identique sur les deux. Néanmoins une telle procédure ignore les différences de performances entre les perceptrons. Par exemple, si une de ces coupures a un excellent taux d'erreur lors de l'apprentissage, une réponse identique sur les deux images lors du test est plus informative que si elle avait eu un taux d'erreur élevé lors de l'apprentissage.

Pour formaliser précisément cette idée, nous proposons de modéliser les réponses des perceptrons avant seuillage $L_1, \ldots, L_M$ à l'aide d'une hypothèse d'indépendance conditionnelle étant donné les vrais labels de l'objet $S_1, \ldots, S_M$ et d'un modèle gaussien des réponses individuelles. Le modèle gaussien de la réponse est motivé

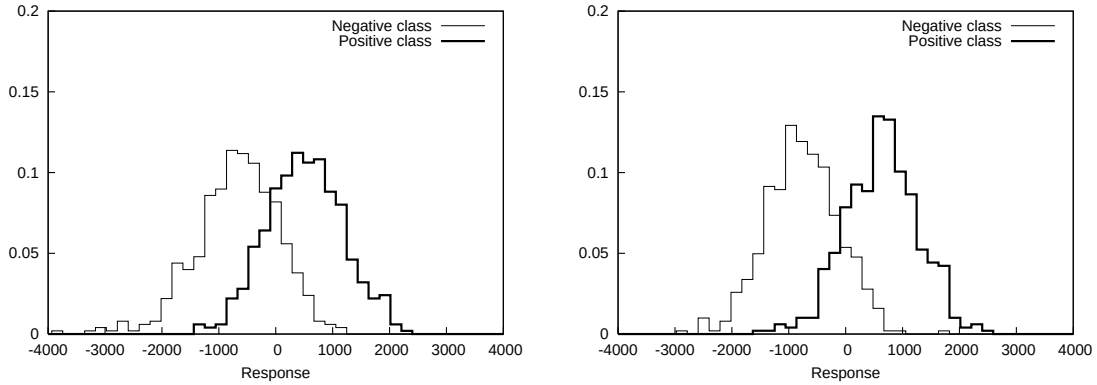

FIG. 2.11 – Ces deux histogrammes sont représentatifs de la réponse des perceptrons avant seuillage conditionnellement au vrai label utilisé lors de l'apprentissage $P(L \mid S)$.

par la forme des distributions empiriques des réponses des perceptrons (figure 2.11), et capture la confiance que l'on peut avoir intuitivement dans une coupure en fonction de ses performances lors de l'apprentissage.

Sous ces hypothèses, nous avons

$$\log \frac{P(C^1 = C^2 \mid \mathbf{L}^1, \mathbf{L}^2)}{P(C^1 \neq C^2 \mid \mathbf{L}^1, \mathbf{L}^2)} = \log \frac{P(\mathbf{L}^1, \mathbf{L}^2 \mid C^1 = C^2)}{P(\mathbf{L}^1, \mathbf{L}^2 \mid C^1 \neq C^2)} + \log \frac{P(C^1 = C^2)}{P(C^1 \neq C^2)} \quad (2.14)$$

et

$$\log \frac{P(\mathbf{L}^1, \mathbf{L}^2 \mid C^1 = C^2)}{P(\mathbf{L}^1, \mathbf{L}^2 \mid C^1 \neq C^2)} = N \log 2 + \sum_i \log \left(\alpha_i^1 \alpha_i^2 + (1 - \alpha_i^1)(1 - \alpha_i^2) \right) , \quad (2.15)$$

où $\alpha_i^j = P(S_i^j = 1 \mid L_i^j)$.

Étant donné notre modèle Gaussien de la réponse des perceptrons conditionnellement au vrai label, la grandeur α_i^j prend la forme d'une sigmoïde appliquée à la réponse du perceptron. Cette sigmoïde d'autant plus étalée que la coupure est ambiguë. Ainsi la règle finale est un comptage pondéré, qui donne plus d'importance aux coupures qui séparent bien les classes lors de l'apprentissage.

Les performances de cette technique sur deux jeux de données (la base d'image COIL-100 et une base de symboles LaTeX bruités) ont montré que nous pouvons atteindre avec $1,000$ perceptrons et un seul exemple une erreur moyenne du même ordre que celle obtenue avec un seul perceptron et entre 16 et 32 exemples.

2.6 Conclusion

Dans les exemples présentés dans ce chapitre, nous avons vu que l'utilisation de modèles génératifs est très naturelle, permet facilement de formaliser une connaissance *a priori* du problème, et surtout permet d'obtenir finalement des règles de prédiction qui sont intuitives et incorporent toutes les composantes, par exemple de régularisation, que l'on sait être importantes.

Cette démarche présente deux avantages pratiques. Elle permet d'une part de préciser où se trouve la faiblesse d'un schéma de prédiction : en imposant de préciser clairement les hypothèses d'indépendance et les modèles de densités, elle permet de préciser quelles seront les situations où la méthode ne fonctionnera pas. D'autre part, elle mène à des règles claires pour combiner des grandeurs et évite de recourir à des termes de régularisation *ad hoc*.

Chapitre 3

Efficacité algorithmique

La composante algorithmique de l'apprentissage statistique a pris ces dernières années une importance croissante. D'une part les données à traiter sont devenues de plus en plus importantes avec l'augmentation des moyens de stockage et d'acquisition (appareils photo numériques en particulier), et d'autre part les techniques mathématiques ont permis de manipuler des objets de plus en plus complexes.

Les travaux présentés dans ce chapitre sont construits sur des techniques algorithmiques rapides. Une partie de ces techniques sont le résultat d'une intégration profonde entre la modélisation statistique et le processus algorithmique (tests séquentiel adaptatifs, théorie du champ moyen), d'autres sont au contraire exactes et relèvent de l'algorithmique classique (images intégrales, évaluation paresseuse).

3.1 Détection de visages

Nous avons vu en §2.2 un modèle de densité d'images de visages reposant sur une hypothèse d'indépendance conditionnelle étant donnée la pose du visage dans le plan image. Nous avons aussi vu que de ce modèle nous pouvons dériver un type de classifieur à deux classes extrêmement simple.

Avec un modèle de ce type, une exploration exhaustive de l'espace des poses possibles amènerait à des résultats optimaux. Néanmoins une telle exploration est algorithmiquement impossible.

La forme de notre algorithme de détection de visages complet a été initialement motivé par l'idée que la relation entre le coût algorithmique et l'efficacité statistique est une relation fondamentale. Au lieu de concevoir un algorithme statistiquement performant, puis de l'optimiser algorithmiquement ensuite, nous avons voulu dés le départ tenir compte de cette relation entre coût et taux d'erreur.

3.1.1 Hiérarchie de classifieurs

Comme décrit en §2.2, le processus de détection consiste à parcourir la scène à traiter en entier, à différentes échelles. Pour chacune de ces positions et de ces échelles, un classifieur à deux classes examine une imagette de taille 64×64 pixels et doit estimer la pose qui maximise la probabilité de l'image examinée sous l'hypothèse "visage" avec notre modèle.

Nous considérons comme l'ensemble de poses admissibles suivant

$$R_0^0 = \{\ (x, y, \theta, s) : 28 \leq x \leq 36,\ 28 \leq y \leq 36,\ -20^\circ \leq \theta \leq 20^\circ,\ 10 \leq s \leq 20\ \} \tag{3.1}$$

que nous décomposons hiérarchiquement (cf. figure 3.1) en cellules R_j^i

$$\forall i \leq D, j \neq k, \qquad R_j^i \cap R_k^i = \emptyset \tag{3.2}$$

$$\forall i < D, \forall j, \exists N, l_1, \ldots, l_N, \qquad R_j^i = \cup_n R_{l_n}^{i+1} \tag{3.3}$$

et construisons pour chaque R_j^i un classifieur

$$f_j^i : \mathcal{I} \to \{0,\, 1\} \tag{3.4}$$

dédié aux images de visages dont les poses sont dans R_j^i. Ce classifieur a une forme similaire à ceux dédiés à des poses exactes, mais le seuil T est choisi de manière à forcer le taux de faux négatifs, c'est à dire la proportion d'images de visages classifiées comme n'étant pas des visages, en dessous d'un seuil fixe. Le taux de faux positifs est quant à lui laissé libre.

Puisque nous disposons de ce modèle explicite, nous pouvons générer des exemples d'apprentissage en forçant la valeur de Γ (simplement en appliquant une translation, une rotation et un facteur d'échelle à l'image, cf. figure 2.6), et ainsi construire pour une valeur donnée de Γ un classifieur dédié de la forme décrite ci-dessus.

Le critère global de détection est l'existence d'une chaîne complète de classifieurs (du sommet de la hiérarchie à une des feuilles) qui répondent positivement

$$\exists a_0, \ldots, a_D, \qquad \forall d < D, R_{a_{d+1}}^{d+1} \subset R_{a_d}^d \tag{3.5}$$

$$\forall d, f_{a_d}^d(I) = 1 \tag{3.6}$$

Dans toutes nos expériences, l'apprentissage du détecteur global, décrit au chapitre 3 a été faite à l'aide d'une base de 400 images de visages, correspondant à 40

personnes chacune prise 10 fois en photos. Ce chiffre est à comparer aux milliers d'images nécessaires pour entraîner une technique discriminative non-supervisée et générique telle qu'une SVM ou une combinaison de classifieurs élémentaires entraînés par boosting.

3.1.2 Évaluation paresseuse

Le fait que ces classifieurs aient un taux de faux négatifs virtuellement nul nous assure que si l'imagette qui est examinée contient effectivement un visage avec une pose Γ, tous les détecteurs associés à des ensembles de poses qui contiennent Γ répondront positivement.

Ainsi, il en découle un processus algorithmique adaptatif : la réponse d'un classifieur associé à un ensemble de poses donné est calculée seulement si les réponses des classifieurs associés à des sur-ensembles ont déjà été calculées et si elles sont positives. Une telle procédure rejette rapidement des zones de l'image qui ne sont pas du tout ambiguës (zones sans structures telles que les murs d'une pièce) et concentre l'effort algorithmique sur des parties de l'image riches en structures ayant l'apparence de visages. La figure 3.2 illustre cette variation du coût en fonction de la complexité de l'image et montre que seules les parties ambiguës et riches en structures demandent un nombre d'opérations élevé.

En faisant une hypothèse raisonnable de convexité de la relation entre le coût d'évaluation d'un classifieur et son taux d'erreur, on peut montrer qu'une telle stratégie est optimale (Fleuret 2000, Jung 2001).

3.2 Sélection de paramètres booléens

De nombreuses techniques d'apprentissage statistique reposent sur une première étape qui consiste à réduire la dimension du signal à traiter tout en conservant l'information relative à la valeur à prédire (Guyon & Elisseeff 2003). Cette première étape peut soit être spécifique à une famille de prédicteur, soit être générique (Kohavi & John 1997, Das 2001).

Nous nous sommes intéressés au développement d'une technique rapide de sélection de paramètres booléens dans un cadre de classification à deux classes. La méthode que nous avons proposée consiste à maximiser l'information mutuelle entre un paramètre à sélectionner et la classe à prédire, conditionnellement à n'importe lequel des paramètres déjà choisis (Vidal-Naquet & Ullman 2003, Fleuret 2003, Fleuret 2004). Cette stratégie assure la sélection de paramètres informatifs et peu redondants, contrairement à une sélection basée par exemple sur l'information mutuelle seule.

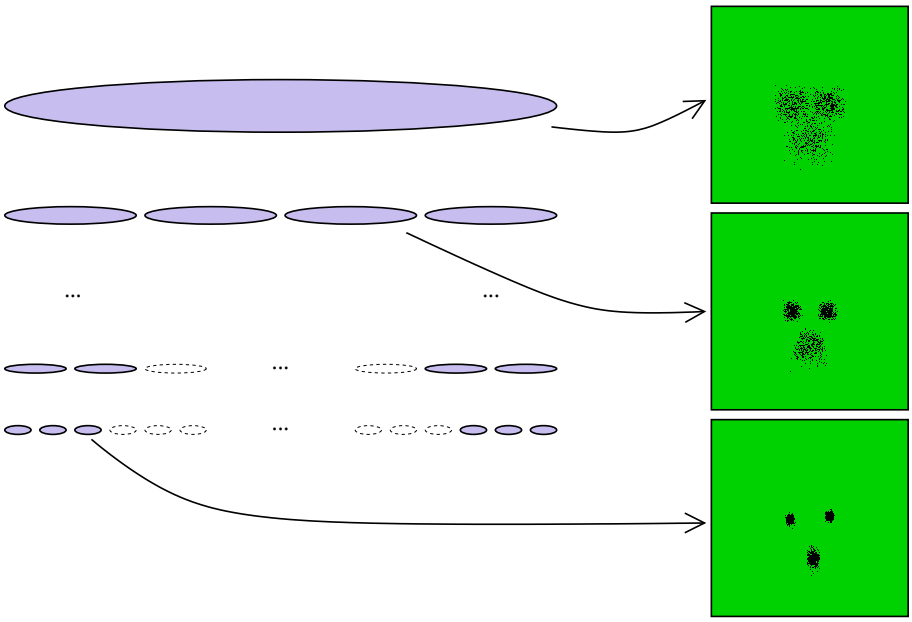

FIG. 3.1 – Hiérarchie de poses de visages.

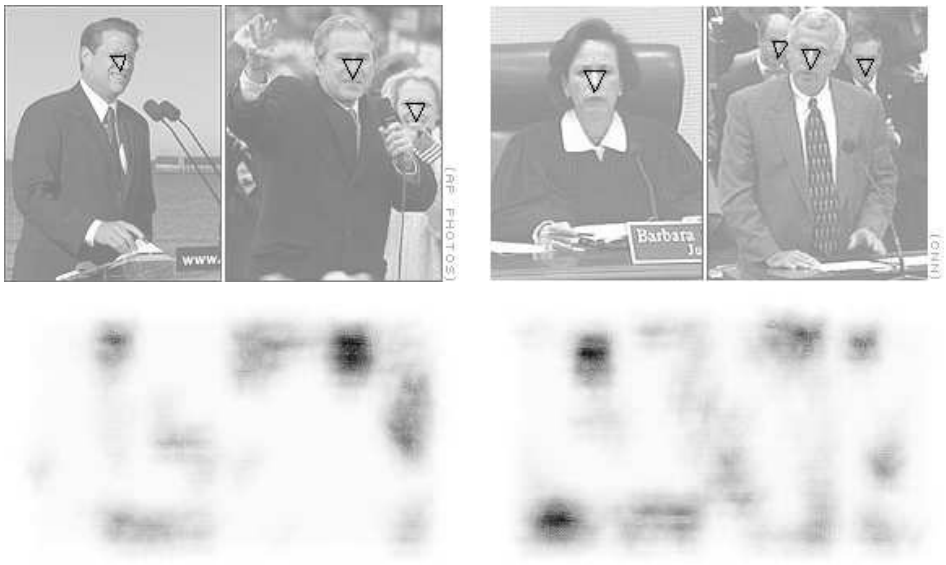

FIG. 3.2 – Résultats de la détection (haut) et intensité algorithmique (bas). L'intensité représentée en niveau de gris est proportionnelle au nombre de fois qu'un pixel est lu lors du processus de détection. Les zones sombres correspondent donc à des zones pour lesquelles un grand nombre de classifieurs ont du être évalués.

Une telle pré-sélection permet de réduire drastiquement le coût algorithmique de l'apprentissage ainsi que le sur-apprentissage. La méthode que nous avons proposée a été utilisée avec succès comme pré-traitement pour des applications telles que la détection de visages (Brubaker et al. 2005) ou la détection de gènes induisant une prédisposition au cancer (Yun & Keong 2005).

3.2.1 Maximisation de l'information mutuelle conditionnelle

Étant donnée une classe à prédire Y et une famille de paramètres booléens $X_1, \ldots, X_N$, où N est de l'ordre de plusieurs dizaines de milliers, l'objectif de la sélection de paramètres est de choisir une sous-famille $X_{\nu(1)}, \ldots, X_{\nu(K)}$ où $K \ll N$ telle que $H(Y \mid X_{\nu(1)}, \ldots, X_{\nu(K)}) \simeq H(Y \mid X_1, \ldots, X_N)$.

Les méthodes classiques consistent à trier les X_n selon un score tel que l'information mutuelle $I(Y; X_n)$, et à ne conserver que les K premiers (Battiti 1994, Bonnlander & Weigend 1996, Torkkola 2003).

Un tel critère est très dangereux lorsque les X_n sont constitués de sous-familles très homogènes dont l'une est beaucoup plus informative que les autres. Dans un tel cas, les K paramètres choisis seront très similaires (on peut imaginer le cas limite où il y a K paramètres identiques et très informatifs, qui seraient donc les seuls sélectionnés).

Le schéma que nous proposons consiste à tenir compte de la redondance entre les paramètres déjà choisis et les nouveaux candidats en utilisant l'information mutuelle conditionnelle entre un nouveau paramètre et la classe à prédire, conditionnellement à chacun de paramètres déjà choisis. La procédure de sélection est la suivante

$$\nu(1) = \arg\max_n \hat{I}\,(Y\ ;\ X_n) \tag{3.7}$$

$$\forall k,\ 1 \leq k < K, \quad \nu(k+1) = \arg\max_n \underbrace{\left\{ \min_{l \leq k} \hat{I}\left(Y\ ;\ X_n \,|\, X_{\nu(l)}\right) \right\}}_{s(n,k)}. \tag{3.8}$$

La grandeur $\hat{I}\left(Y\ ;\ X_n \,|\, X_{\nu(l)}\right)$ est petite si X_n ne porte aucune information sur Y ou bien si cette information avait déjà été capturée par $X_{\nu(l)}$. Le score $s(n,k)$ est grand uniquement si $X_{\nu(k)}$ porte une information qui n'avait été capturée par aucun des paramètres choisis précédemment.

Bien que cette procédure ne protège pas des dépendances de degré supérieure, par exemple entre triplés de variables, les performances mesurées empiriquement

montrent que combinée avec une technique de classification aussi simple que le Bayésien naïf, elle est comparable à des techniques telles que le boosting ou les SVMs.

3.2.2 Organisation rapide du calcul

En notant M le nombre d'exemples d'apprentissage, le calcul de $\hat{I}\left(Y\ ;\ X_n \,|\, X_{\nu(l)}\right)$ coûte $O(M)$. Donc la procédure (3.7) induit un coût algorithmique $O(K \cdot N \cdot M)$. Ce coût est rédhibitoire pour des problèmes avec plusieurs dizaines de milliers de paramètres et d'exemples d'apprentissage.

On peut néanmoins remarquer que le score $s(n,k)$ étant le minimum d'un série de termes de plus en plus longue, il ne peut que diminuer.

La version rapide de ce schéma consiste à tenir à jour une table $s(n, m(n,k))$ de scores partiels, chacun calculé sur les $m(n,k) \leq k$ premiers termes de l'opérateur de minimum de l'équation (3.8). Le rang $m(n,k)$ est le plus petit rang tel que le score partiel $s(m, m(n,k))$ soit plus petit que le meilleur score exact calculé jusque là, et k s'il est le meilleur.

$$m(0,k) = k \tag{3.9}$$

$$m(n,k) = \min\left\{ l : l \leq k,\ s(n,l) \leq \max_{o \leq n} s(o, m(o,k)) \right\} \tag{3.10}$$

Avec une telle procédure, le nombre d'évaluations de $\hat{I}\left(Y\ ;\ X_n \,|\, X_{\nu(l)}\right)$ est expérimentalement divisé par près de 100, avec un gain en vitesse équivalent (cf. figure 3.3).

Finalement, cette technique peut sélectionner 50 paramètres parmi $40,000$, à partir d'un ensemble d'apprentissage de 500 exemples en un dixième de seconde sur un ordinateur PC équipé d'un processeur à 1Ghz.

3.3 Suivi de personnes multi-caméras

Nous avons présenté en §2.3 un modèle d'apparence de l'image produite par un algorithme de segmentation basé sur le mouvement conditionnellement à l'occupation du sol. De ce modèle, nous dérivons un large système d'équations dont la solution approxime les probabilités marginales d'occupation conditionnellement au signal.

Le point faible de ce schéma est son coût algorithmique. Le système d'équations à résoudre comporte autant d'inconnues qu'il y a de positions au sol dans la

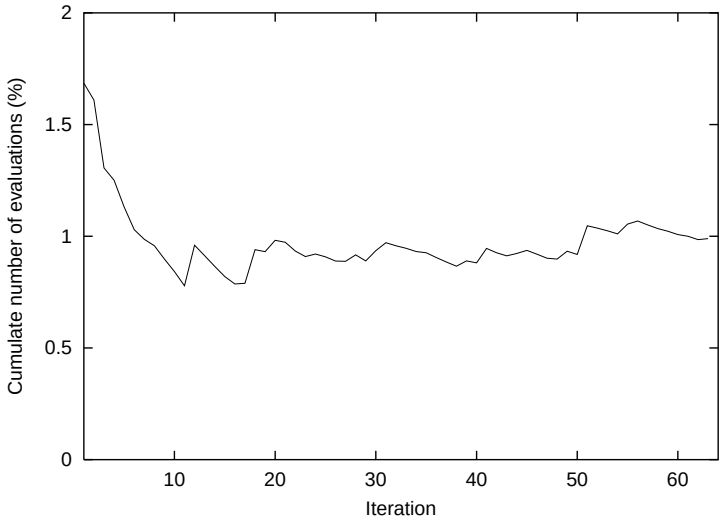

FIG. 3.3 – Cette courbe montre le ratio entre les nombres d'évaluations d'informations mutuelles conditionnelles dans le cas d'un algorithmes simple et dans le cas paresseux proposé.

discrétisation utilisée, chaque équation fait intervenir une espérance conditionnelle très lourde à calculer numériquement, et le système lui-même n'étant pas linéaire, le résoudre demande plusieurs dizaines d'itérations.

3.3.1 Linéarisation de l'espérance conditionnelle

Nous avons vu en §2.3.3 que l'équation principale à résoudre (3.13) demande en particulier d'évaluer deux termes de la forme

$$E_Q\left(\Psi(B^c, A^c) \mid X_n = \xi\right) \tag{3.11}$$

où $\Psi(B^c, A^c) = \frac{\|B^c - A^c\|}{\sigma \|A^c\|}$. Par définition de Q, sous cette loi, l'image A^c est concentrée autour de la vraie image acquise B^c. Nous linéarisons donc Ψ pour approximer (3.11) par

$$\Psi(B^c, E_Q(A^c \mid X_n = \xi)) \tag{3.12}$$

Une telle approximation est classique en physique, et correspond à la technique du champ moyen. Cela nous amène à la forme finale du large système d'équations à résoudre :

$$q_k = \frac{1}{1 + \exp\left(\lambda_k + \sum_c \Psi(B^c, E_Q(A^c \,|\, X^k = 1)) - \Psi(B^c,\, E_Q(A^c \,|\, X^k = 0))\right)} \tag{3.13}$$

Puisque la loi Q est une loi produit, en notant $q_n = Q(X_n = 1)$ les marginales et $\mathcal{A}_n$ la silhouette rectangulaire correspondant à la position n vue de la caméra c, nous avons

$$\begin{aligned}
\forall x, y,\ E_Q(A^c(x,y)) &= Q(A^c(x,y) = 1) && (3.14)\\
&= 1 - Q(\forall k, \mathcal{A}_k^c(x,y)\, X_k = 0) && (3.15)\\
&= 1 - \prod_{k:\mathcal{A}_k^c(x,y)=1} (1 - q_k) && (3.16)
\end{aligned}$$

Le conditionnement par $X_n = \xi$ débouche sur la même expression en substituant 1 ou 0 à la marginale q_n.

3.3.2 Estimation rapide de Ψ

Finalement, l'estimation d'un terme de la forme (3.13) requiert le calcul de deux normes L^1 entre des images, ce qui a un coût algorithmique proportionnel à la résolution de l'image, rédhibitoire en pratique.

Néanmoins, nous pouvons organiser le calcul de façon à ne devoir faire que des intégration des pixels de $E_Q(A^c)$ sur des régions rectangulaires $\mathcal{A}_n^c$, or cela peut être fait à temps constant en pré-calculant l'image intégrale (Simard et al. 1999)

$$\forall x, y,\ I^c(x,y) = \sum_{\alpha \le x, \beta \le y} E_Q(A^c)(\alpha, \beta) \tag{3.17}$$

qui nous donne

$$\begin{aligned}
\sum_{x,y \in \mathcal{A}_n^c} E_Q(A^c)(x,y) &= I^c(x_{\min}(\mathcal{A}_n^c),\, y_{\min}(\mathcal{A}_n^c)) && (3.18)\\
&+ I^c(x_{\max}(\mathcal{A}_n^c),\, y_{\max}(\mathcal{A}_n^c)) && (3.19)\\
&- I^c(x_{\min}(\mathcal{A}_n^c),\, y_{\max}(\mathcal{A}_n^c)) && (3.20)\\
&- I^c(x_{\max}(\mathcal{A}_n^c),\, y_{\min}(\mathcal{A}_n^c)) && (3.21)
\end{aligned}$$

Comme $E_Q(A^c \,|\, X_n = 1)$ et $E_Q(A^c)$ ne diffèrent que sur $\mathcal{A}_n^c$ nous avons par exemple

$$
\begin{aligned}
&\|E_Q(A^c \mid X_n = 1)\| \\
&\quad = \sum_{x,y} E_Q(A^c \mid X_n = 1)(x,y) \\
&\quad = \sum_{x,y} E_Q(A^c)(x,y) + \sum_{x,y \in \mathcal{A}_n^c} E_Q(A^c \mid X_n = 1)(x,y) - E_Q(A^c)(x,y) \\
&\quad = \sum_{x,y} E_Q(A^c)(x,y) + \sum_{x,y \in \mathcal{A}_n^c} 1 - E_Q(A^c)(x,y) \\
&\quad = \sum_{x,y} E_Q(A^c)(x,y) + \|\mathcal{A}_n^c\| - \sum_{x,y \in \mathcal{A}_n^c} E_Q(A^c)(x,y)
\end{aligned}
\tag{3.22}
$$

et tous ces termes peuvent être calculés avec un coût algorithmique constant.

Un algorithme décrit en détail dans (Fleuret et al. 2006) généralise cette stratégie à tous les termes de (3.13) et permet d'obtenir un algorithme capable de traiter une vidéo à raison de six frames par seconde sur un PC à 2.4Ghz.

La figure 3.4 montre un des résultats intermédiaires lors de la convergence de l'algorithme sur une séquence à quatre caméras. Les performances de cet algorithme seul, n'ayant accès ni signal vidéo complet ni à la cohérence temporelle de mouvement, sont de l'ordre de 6.5% d'erreur de faux négatifs (personne présente non détectée) et de 4% de faux positifs (Fleuret et al. 2005). Une fois combiné avec un modèle complet de mouvement et d'apparence reposant sur la couleur, ce taux d'erreur est nul sur les séquences de test que nous avons utilisées, et l'erreur métrique est 80% du temps inférieure à 25cm (Berclaz et al. 2006).

3.4 Conclusion

Nous avons montré dans ce chapitre que l'utilisation de techniques algorithmiques efficaces permet de manipuler des modèles mathématiquement complexes tout en conservant une vitesse de traitement utilisable.

La détection hiérarchique de visages et la sélection rapide de paramètres reposent toutes les deux sur une procédure adaptative et paresseuse : en fonction des données rencontrées, elles limitent le calcul à un sous-ensemble nécessaire d'évaluations, ce qui réduit de plusieurs ordres de magnitudes le nombre d'opérations nécessaires.

Le suivi de personnes repose sur une construction algorithmique plus classique, et exacte. Néanmoins, cette organisation débouche sur un algorithme très rapide, et extrêmement robuste au bruit.

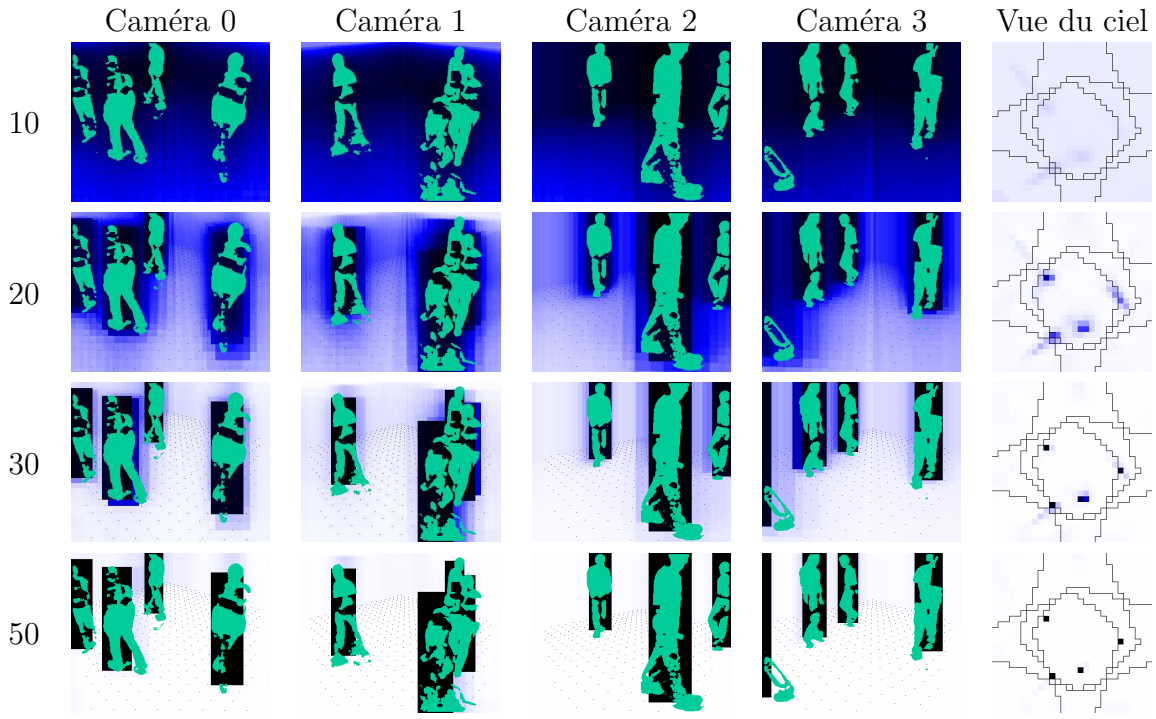

FIG. 3.4 – Chaque ligne correspond à une itération du système d'équations (3.13). Les nuances de bleus correspondent dans les vues caméras aux valeurs des pixels dans les images moyennes $E_Q(A^c)$ et dans les vues du ciel (colonne de droite) aux probabilités marginale d'occupation q_n.

Que ces procédure soient exactes, comme dans le cas de la sélection de paramètres ou de suivi de personnes, ou qu'elles soient des approximations comme dans le cas de la détection de visages, l'efficacité algorithmique qui en découle est une condition indispensable pour une utilisation opérationnelle.

Bibliographie

Amari, S. & Wu, S. (2000), 'Improving support vector machine classifiers by modifying kernel functions', *Neural Networks* **12**, 783 – 789.

Amit, Y. (2002), *3D Object Detection and Recognition*, MIT Press.

Battiti, R. (1994), Using mutual information for selecting features in supervised neural net learning, *in* 'IEEE Transactions on Neural Networks', Vol. 5.

Berclaz, J., Fleuret, F. & Fua, P. (2006), Robust people tracking with global trajectory optimization, *in* 'Proceedings of the IEEE international conference on Computer Vision and Pattern Recognition (CVPR)', Vol. 1, pp. 744–750.

Bonnlander, B. V. & Weigend, A. S. (1996), Selecting input variables using mutual information and nonparametric density estimation, *in* 'Proceedings of ISANN'.

Brubaker, C., Wu, J., Sun, J., Mullin, M. & Rehg, J. (2005), On the design of cascades of boosted ensembles for face detection, Technical Report GIT-GVU-05-28, Georgia Institute of Technology.

Christiani, N. & Shawe-Taylor, J. (2000), *An Introduction to Support Vector Machines and other kernel-based learning methods*, Cambridge University Press.

Das, S. (2001), Filters, wrappers and a boosting-based hybrid for feature selection, *in* 'Proceedings of ICML2001', pp. 74–81.

Duda, R. & Hart, P. (1973), *Pattern classification and scene analysis*, John Wiley & Sons.

Fleuret, F. (2000), Détection hiérarchique de visages par apprentissage statistique, PhD thesis, Université Paris-VI, Paris.

Fleuret, F. (2003), Binary feature selection with conditional mutual information, Technical Report RR-4941, INRIA.

Fleuret, F. (2004), 'Fast binary feature selection with conditional mutual information', *Journal of Machine Learning Research (JMLR)* **5**, 1531–1555.

Fleuret, F. & Blanchard, G. (2005), Pattern recognition from one example by chopping, *in* 'Proceedings of the Neural Information Processing Systems Conference (NIPS)', pp. 371–378.

Fleuret, F. & Geman, D. (1999), Graded learning for object detection, *in* 'Proceedings of the workshop on Statistical and Computational Theories of Vision of the IEEE international conference on Computer Vision and Pattern Recognition (CVPR/SCTV)'.

Fleuret, F. & Geman, D. (2000), Apprentissage hiérarchique pour la détection de visages, *in* 'Proceedings of the French conference on Pattern Recognition and Artificial Intelligence (RFIA)', Vol. 2, pp. 349–357.

Fleuret, F. & Geman, D. (2001), 'Coarse-to-fine face detection', *International Journal of Computer Vision (IJCV)* **41**(1/2), 85–107.

Fleuret, F. & Geman, D. (2002), Fast face detection with precise pose estimation, *in* 'Proceedings of the IEEE International Conference on Pattern Recognition (ICPR)', Vol. 1, pp. 235–238.

Fleuret, F. & Gerstner, W. (2005), A Bayesian kernel for the prediction of neuron properties from binary gene profiles, *in* 'Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)', pp. 129–134.

Fleuret, F., Berclaz, J., Lengagne, R. & Fua, P. (2006), Multi-camera people tracking with a probabilistic occupancy map, Technical Report EPFL/CVLAB2006.06, EPFL.

Fleuret, F., Lengagne, R. & Fua, P. (2005), Fixed point probability field for complex occlusion handling, *in* 'Proceedings of the IEEE International Conference on Computer Vision (ICCV)', Vol. 1, pp. 694–700.

Guyon, I. & Elisseeff, A. (2003), 'An introduction to variable and feature selection', *Journal of Machine Learning Research* **3**, 1157–1182.

Hamill, O. P., Marty, A., Neher, E., Sakmann, B. & Sigworth, F. J. (1981), 'Improved patchclamp techniques for high-resolution current recording from cells and cell-free membrane patches', **391**, 85–100.

Jung, F. (2001), Reconnaissance d'objets par focalisation et détection de changements, PhD thesis, École Polytechnique.

Kohavi, R. & John, G. (1997), 'Wrappers for feature subset selection', *Artificial Intelligence* pp. 273–324.

Langley, P., Iba, W. & Thompson, K. (1992), An analysis of bayesian classifiers, *in* 'Proceedings of AAAI-92', pp. 223–228.

Ng, A. & Jordan, M. (2002), On discriminative vs. generative classifiers : A comparison of logistic regression and naive bayes, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)'.

Rowley, H., Baluja, S. & Kanade, T. (1998), Rotation invariant neural network-based face detection, *in* 'Proceedings of IEEE Conference on Computer Vision and Pattern Recognition'.

Sakmann, B. & Neher, E. (1983), *Single Channel Recording*, Plenum.

Simard, P., Bottou, L., Haffner, P. & Cun, Y. L. (1999), Boxlets : a fast convolution algorithm for signal processing and neural networks, *in* 'Advances in Neural Information Processing Systems', Vol. 11, pp. 571–577.

Smola, A. J. & Schölkopf, B. (1998), A tutorial on support vector regression, Technical Report NC-TR-98-030, NeuroCOLT, Royal Holloway College, University of London.

Toledo-Rodriguez, M., Blumenfeld, B., Wu, C., Luo, J., Attali, B., Goodman, P. & Markram, H. (2004), 'Correlation maps allow neuronal electrical properties to be predicted from single-cell gene expression profiles in rat neocortex', *Cerebral Cortex* **14(12)**, 1310–1327.

Torkkola, K. (2003), 'Feature extraction by non-parametric mutual information maximization', *Journal of Machine Learning Research* **3**, 1415–1438.

Ulusoy, I. & Bishop, C. M. (2005), Generative versus discriminative methods for object recognition, *in* 'Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition', Vol. 2, pp. 258–265.

Vapnik, V., N. (1998), *The Nature of Statistical Learning Theory*, Springer-Verlag.

Vidal-Naquet, M. & Ullman, S. (2003), Object recognition with informative features and linear classification, *in* 'Proceedings of ICCV2003', pp. 281–288.

Viola, P. & Jones, M. (2001), Robust real-time object detection, Technical Report 1, Compaq Cambridge Research Lab.

Yun, Z. & Keong, K. C. (2005), Identifying simple discriminatory gene vectors with an information theory approach, *in* 'Proceedings of the Computational Systems Bioinformatics Conference', pp. 13–24.