N° : 2005-21 **ANNEE : 2005**

THESE

POUR OBTENIR LE GRADE DE

DOCTEUR DE L'ECOLE CENTRALE DE LYON

DISCIPLINE : INFORMATIQUE

(arrêté du 30 mars 1992)

PRESENTEE PAR

Mohamed HAMMAMI

Modèle de peau et application à la classification d'images et au filtrage des sites Web

DIRECTEUR DE THESE **: Liming CHEN**

JURY

M. Jacques LE MAITRE	Professeur - Université de Toulon et du VAR	
M. Mesaac MAKPANGOU	Professeur - INRIA, Rocquencourt	Rapporteur
Mme Florence SEDES	Professeur - Université Paul Sabatier, Toulouse	Rapporteur
M. Abdelmajid BEN HAMADOU	Professeur - Université de Sfax - Tunisie	Examinateur
M. Liming CHEN	Professeur - Ecole Centrale de Lyon	Directeur de Thèse
M. Youssef CHAHIR	Maître de conférences - Université de Caen	Co-encadrant

Remerciements

J’adresse mes remerciements à toutes les personnes qui m’ont soutenu lors de la réalisation de ce travail de thèse :

À mon directeur de thèse Monsieur Liming CHEN, Professeur à l’Ecole Centrale de Lyon, pour m’avoir encadré et guidé en me ramenant sur le juste chemin au cours de mes travaux. Ses commentaires et ses suggestions sont et seront toujours appréciés. Toute ma reconnaissance lui adressée ici non seulement pour sa disponibilité, mais également pour la confiance qu’il m’a témoignée, pour son soutien et sa compréhension. Qu’il soit persuadé de mon profond respect.

À M. Youssef CHAHIR, co-encadrant de cette thèse, pour l’attention et l’amitié qu’il m’a apportée durant ma thèse. Je lui suis très reconnaissant pour son aide, ses conseils, ses qualités humaines et ses encouragements.

À M. Jacques LE MAITRE, Professeur à l’Université de Toulon pour avoir accepté de présider cette thèse.

À M. Mesaac MAKPANGOU, Professeur à l’INRIA Rocquencourt et Mme Florence SEDES, Professeur à l’Université Paul Sabatier de Toulouse pour avoir accepté de rapporter ce travail de thèse.

À M. Abdelmajid BEN HAMADOU, Directeur de l’Institut Supérieur d'Informatique et du Multimédia de Sfax (Tunisie) pour avoir examiner ce travail.

À Monsieur Djamel ZIGHED, Directeur du laboratoire ERIC, pour ses conseils, son amabilité sa disponibilité et son aide surtout au début de ma thèse. Qu'il trouve ici l'expression de ma profonde gratitude.

À tous les membres de l’équipe qui ont montré leur amitié et leur encouragement : M. Mohsen ARDABILIAN, Mme Colette VIAL BULL, M. Alexandre SAIDI, M. Emmanuel DELLANDREA, M. Mohand MOUSAOUI, M. René CHALON, Mme Françoise CHATELIN et plus particulièrement M. Christian VIAL pour sa remarquable disponibilité, sa gentillesse, ainsi que les multiples conseils qu’il m’a prodigués au cours de nos nombreuses discussions.

Au directeur de LIRIS, M. Bernard PEROCHE, au Directeur du Département Mathématique-Informatique M. Jean-Pierre LOHEAC ainsi qu’à tous les membres du département.

À mes amis Riadh, Sondes, Walid, Dzmitry, Hadi et Alain qui m’ont soutenu et avec qui j’ai passé de bons moments de détente durant ces dernières années. Je remercie plus particulièrement Mohamed Ali et Boulbaba qui ont partagé tous les moments de cette thèse en plus de la cohabitation dans le même appartement et qui ont su me supporter au quotidien, surtout ces derniers temps.

Enfin, je suis heureux d’associer à ce travail mes parents, mes frères, et mon ami Walid MAHDI pour leurs encouragements et leur soutien.

Tables des matières

Chapitre 1

Introduction

A. Contexte

L'augmentation régulière de la puissance de calcul des microprocesseurs, les progrès réalisés dans les méthodes de développement des logiciels, la numérisation du son, puis des images (i.e., le multimédia), la compression des signaux et le déploiement des réseaux accélérés par les fibres optiques et les satellites dessinent de plus en plus clairement les contours d'un nouveau paysage : la Société de l'Information. Cette société est construite autour de voies électroniques ou de réseaux chargés d'acheminer dans les entreprises, les universités, les administrations, les écoles et les maisons une palette très large de services interactifs. Parmi ces derniers, on peut citer la messagerie, la visiophonie, la télé-enseignement, la consultation de banques de données, la télé-achat, la télévision à la demande, etc. bouleversant de manière radicale et irréversible la vie des individus et des institutions.

Avec le développement du multimédia se prépare sans doute la remise en cause de la communication écrite telle que nous la concevons depuis Gutenberg. De nos jours les lecteurs doivent apprendre à naviguer dans un monde d'images et de textes. Le simple « clic » sur un mot ou une image permet un accès, quasi instantané, à n'importe laquelle des informations stockées dans l'un des ordinateurs de la planète. Enfin, le développement des réseaux du futur va changer le fonctionnement même de la société modifiant par exemple l'organisation du travail, l'accès des citoyens aux services de santé ou de l'éducation, les relations administration/administré, voire les conditions d'exercice de la démocratie.

B. Problématique et Objectifs

Une information de plus en plus visuelle est une conséquence majeure de cette convergence entre l'informatique, Internet et l'audiovisuel. De plus en plus d'applications produisent, utilisent et diffusent des données visuelles, incluant des images fixes et animées. L'augmentation significative des informations visuelles au sein de Internet et dans les organisations s'est accompagnée d'une prise de conscience de l'importance de développer des moyens informatiques pour traiter ces informations. Ce traitement permet de modéliser, de filtrer, de classer, de rechercher et d'indexer cette quantité importante de données. Si la numérisation d'image est déjà un problème techniquement résolu, il n'est pas de même avec la classification, le catalogage et l'indexation d'image. Dans un tel contexte, il est impératif de pouvoir classifier les images selon leurs thèmes ou leurs contenus. Cette classification permettra de faire une sélection ou un contrôle d'accès selon la sémantique et selon le type des images. Ce travail s'effectue actuellement de manière manuelle par annotation, ce qui présente un coût en temps et en main d'œuvre trop important. De plus, la taille des collections d'images est de plus en plus gigantesque, ce qui rend leur annotation manuelle quasi impossible. En conséquence, nous nous trouvons actuellement devant une masse d'informations visuelles mal classifiée et incontrôlée sur Internet.

C'est précisément dans ce contexte que s'inscrivent les travaux que nous avons développés dans le cadre de cette thèse. Nous nous sommes principalement intéressés au problème d'identification de pixels de peau dans l'image. Ce sujet de recherche est d'enjeu important dans la mesure où il est indispensable avant d'envisager des analyses et des traitements de niveau supérieur. Notre objectif est de construire un modèle de peau le plus générique possible qui servira ensuite de base pour le développement d'un système de

détection des parties du corps humain (visage, main, etc.) qui sera d'une utilité potentielle pour de nombreuses applications telles que les applications liées à la sécurité (surveillance, contrôle d'accès, etc.). Outre l'identification, un système de détection et de reconnaissance de visages peut également être utilisé pour faciliter la recherche et la navigation dans une masse de vidéos. La mise en place d'un tel système doit, au préalable, s'appuyer sur un système de détection et de segmentation des régions de peau. Une autre application immédiate de notre approche est le filtrage des images sur le Web. En effet, Internet apparaît comme un immense gisement d'informations dont le libre accès conduit à des usages indésirables tel que l'accès des enfants à des sites adultes. Il est donc nécessaire d'introduire des outils de filtrage de sites pour des applications comme le contrôle parental. Ces outils doivent s'appuyer sur une analyse sémantique d'images dans le processus d'identification où le texte à lui seul n'est plus suffisant.

Dans la littérature, il existe de nombreuses méthodes pour détecter dans l'image les régions de peau, comme le visage et/ou les mains d'une personne. Trois approches principales se détachent : l'une, basée sur la couleur, l'autre, basée sur l'extraction de traits caractéristiques et la dernière basée sur le mouvement. Notre objectif est de développer un modèle de peau permettant de détecter toutes les régions de peau d'une image en un temps raisonnable. De ce fait, nous avons choisi une méthode basée sur la détection de peau par une approche couleur.

De nombreux travaux ont été réalisés sur l'élaboration d'un modèle de peau basé sur l'indice couleur. La difficulté réside dans la prise en compte des conditions de lumière, de la richesse ethnique avec des teintes variées et des décors complexes. En effet, la plupart des travaux pratiquent une phase d'apprentissage sur des classes prédéfinies d'images, sous des conditions d'éclairage connues à l'avance. Si ces modèles conviennent généralement aux systèmes à base d'images peu variées, ils sont peu adaptés aux systèmes contenant une grande variété d'images.

C. Notre démarche et nos contributions

Les grands volumes de bases de données, la diversité et l'hétérogénéité des sources de données nécessitent une nouvelle philosophie de traitement de celles-ci. Dans ce contexte, la fouille de données (Data mining) s'intéresse à découvrir des connaissances implicitement contenues dans un ensemble de données en s'appuyant sur différentes techniques qui peuvent être mises en œuvre indépendamment ou couplées. Ces techniques visent à explorer les données, à décrire leur contenu et à en extraire l'information la plus significative. Parce qu'une grande partie de l'information qui existe dans les organisations est informelle et non structurée, ces techniques ne se limitent pas à des données numériques et factuelles, mais doivent aussi s'adresser aux données textuelles et multimédias. Dans cette thèse, nous introduisons un nouveau champ d'application de la fouille de données aux collections d'images afin de définir un modèle de peau efficace. Pour des raisons diverses telles que les conditions d'éclairage, la diversité ethnique, etc., l'identification de la peau est un problème complexe. Il nous semble intéressant d'appliquer les techniques de data mining dans ce contexte.

Nos premiers travaux, démarrés en 2001, ont permis la définition d'un modèle de peau permettant de discriminer les pixels de peau de ceux de non-peau. Ce modèle est

essentiellement basé sur l'indice couleur, celui-ci étant la primitive la plus riche et la plus simple à calculer. Notre approche trouve son originalité dans le choix des axes de couleurs les plus pertinents, la manière dont ce choix est fixé et l'utilisation de techniques de data mining comme une nouvelle philosophie de traitement des données. Nous aboutissons ainsi à un modèle de peau robuste aux variations de conditions de lumière et à la richesse ethnique. Une amélioration est apportée à nos résultats par l'application d'un nouvel indice : *la distribution spectrale*. Notre hypothèse est que la couleur d'un objet dépend fortement de ses caractéristiques, en particulier, la couleur de peau présente une bande spectrale de largeur déterminée. Les résultats expérimentaux ont montré que la distribution spectrale est d'une importance capitale. Pour cette raison, nous l'avons utilisée comme critère déterminant dans notre processus de classification des pixels de peau. Ces travaux ont été décrits dans [3][5][8].

Afin d'exploiter les règles obtenues à partir de la première phase de notre travail, il était nécessaire d'identifier les régions de couleur de peau dans l'image. Notre contribution consiste à l'élaboration d'une méthode de segmentation souple qui répond à nos exigences en matière de cohérence et d'homogénéité selon différents critères tout en tenant compte de notre modèle de peau. Ce travail a fait l'objet d'une communication dans une revue [1].

Enfin, nous avons réalisé deux applications dans lesquelles la détection et la classification de pixels est une étape préalable nécessaire. La première est la classification des images en gros plan, plan américain et plan en pied réalisée dans le cadre du projet RNTL MUSE (Multimédia Search Engine)[9]. Ce projet a été conduit en collaboration avec les universités de Versailles et de Toulon et en association avec une start-up parisienne E-XMLmedia et un utilisateur final de photos (l'agence Editing). L'objectif du projet MUSE était la réalisation d'un moteur de recherche multimédia sur le Web. La deuxième application est celle de filtrage de sites adultes sur Internet. Ce dernier travail a permis de mettre au point un logiciel, appelé WebGuard, permettant un filtrage sémantique du contenu Web et qui fait actuellement l'objet d'un transfert de technologie à une société de moteur de recherche.

WebGuard est un logiciel innovant basé sur une gestion des contenus visuels et textuels permettant l'analyse et le filtrage du contenu à caractère litigieux sur Internet. L'innovation de la technologie au coeur du logiciel WebGuard provient essentiellement de la combinaison judicieuse des technologies de data mining et d'analyse d'images. Ce travail a fait l'objet de plusieurs publications dans des conférences internationales [6] [7] [10] et dans une revue internationale [1].

L'ensemble des travaux ont donné lieu à :

- 1 revue internationale
- 1 revue internationale IEEE (en cours)
- 1 revue nationale
- 9 conférences internationales avec actes et comité de lecture
- 1 poster

D. Organisation de la thèse

L'organisation du présent manuscrit est présentée ci-dessous.

Le chapitre 2 est consacré à l'état de l'art sur les travaux d'identification des pixels de peau dans l'image. Les différentes méthodes ont été classées et présentées en trois approches. Leurs avantages et défauts sont discutés.

Le chapitre 3 aborde le principe et le processus de data mining qui est à la base de nos travaux. Nous détaillons les algorithmes utilisés au cours de notre travail et nous décrivons les techniques d'évaluations des classifieurs.

Le chapitre 4 est dédié à la définition d'un modèle de peau. Nous proposons une technique de détection et de segmentation de zones de couleur de peau. Cette méthode s'appuie sur des techniques de data mining et d'analyse d'images permettant de définir un modèle de peau capable de différencier les pixels de peau de ceux de non-peau suivant différents axes. La méthode utilise les techniques de data mining pour produire des règles de prédiction. Ces dernières seront utilisées par la suite dans une phase de segmentation de l'image en régions cohérentes de peau.

Le chapitre 5 est consacré à l'application principale de notre travail : la technique de filtrage de sites adulte WebGuard. L'innovation de la technologie au coeur du logiciel WebGuard provient essentiellement de la combinaison judicieuse des technologies de data mining et d'analyse d'images. Pour appréhender le caractère innovant de cette technologie, nous présentons la chronologie des travaux effectués et nous montrons pourquoi la technologie développée est particulièrement adaptée et optimale pour le filtrage des sites Web.

Enfin, le dernier chapitre comporte les conclusions et quelques réflexions sur les ouvertures possibles et sur nos travaux futurs.

Chapitre 2

Etat de l'art : Modèle de peau

CHAPITRE 2 7

ETAT DE L'ART : MODELE DE PEAU 7

2.1 Introduction

La détection de peau consiste à détecter les pixels correspondant à une peau humaine dans une image couleur. La sortie d'un système mettant en œuvre ce principe, est une image binaire ayant la même taille que l'image d'entrée avec une valeur pour la peau et une autre valeur pour l'arrière plan. La figure 2.1 montre une image couleur et son masque binaire après détection des régions de peau.

Figure 2. 1. *(a) et (b) représentent respectivement l'image originale et son masque binaire.*

La détection de peau dans les images couleur est une technique importante utilisée dans de nombreux problèmes de reconnaissance de formes. Elle est souvent une étape préliminaire indispensable pour des problèmes tels que :

- la détection de visages qui composent la majorité des systèmes utilisant ce type d'algorithmes [67] [64] [95] [68] [71] [84];
- la détection des mains [24] [116];
- ou encore la détection de corps humains nus dans une image [92] [93].

La résolution partielle de ces problèmes permet de réaliser des progrès dans les applications suivantes:

- la recherche des images sur le web [86] [87] ;
- le filtrage du web, afin, par exemple, de signaler les images adultes ;
- l'indexation des images par le contenu ;
- la segmentation de la vidéo [88] ;
- ou encore la visioconférence, en permettant le suivi des visages dans des séquences d'images [23] [94] [96] [97].

La plupart des algorithmes de détection de peau ont été conçus pour des applications particulières, et leur évaluation est souvent basée sur l'efficacité du système entier et non seulement sur la partie de détection de peau dans le système.

La détection de peau n'est pas une tâche facile pour des raisons diverses telles que les conditions d'éclairage généralement inconnues (figure 2.2), la diversité ethnique avec des teintes qui varient selon la personne et selon les différentes races (figure 2.3), sans oublier

qu'un décor complexe peut compliquer cette tâche (figure 2.4). La détection de peau est encore plus difficile pour les images issues du Web qui sont capturées sous diverses conditions de lumière et avec différents dispositifs ayant des caractéristiques spécifiques.

Figure 2. 2. *Différents éclairages*

Figure 2. 3. *Diversité ethnique*

Figure 2. 4. *Décors complexes*

Il existe de nombreuses méthodes pour détecter les régions de peau dans une image. On en identifie trois approches majeures que nous développons dans la suite :

- la détection basée sur l'extraction de traits caractéristiques du visage et /ou des mains [14] [15] ;
- La détection basée sur la couleur [13] [67] ;
- et la détection basée sur le mouvement. Cette dernière n'est jamais utilisée seule pour la détection, on la trouve utilisée conjointement avec l'information de couleur de peau dans [60] et couplée avec un système basé sur la reconnaissance de traits caractéristiques dans [61] [62].

2.2 Approches basées sur la géométrie et l'extraction de traits caractéristiques

Les méthodes basées sur la géométrie et l'extraction de traits caractéristiques sont généralement appliquées pour la segmentation des régions de peau des visages et/ou des mains. Cette segmentation est utile pour la reconnaissance des personnes et des gestes : elle permet d'identifier une personne, de la différencier des autres intervenants, ou d'interpréter l'un de ses gestes. Une telle segmentation nécessite généralement la définition de modèles pour les traits caractéristiques du modèle cherché, et parfois même un modèle du corps humain plus ou moins détaillé.

Dans le cas de la détection de visage par exemple, on recherche souvent des traits caractéristiques tels que les yeux, les contours extérieurs, le nez et la bouche que l'on associe à des configurations ("templates") connues a priori ou apprises [45]. Avec l'adjonction de

contraintes géométriques, il a été possible de définir le positionnement relatif de ces traits caractéristiques et d'en obtenir une information sur la présence ou non d'un visage. Ainsi, le système proposé par Govindaraju *et al.* [47], basé sur le contour, modélise le visage comme étant un agencement de trois courbes (sommet, coté droit et coté gauche du visage). La détection de ces courbes et leur regroupement selon leur position relative permet de connaître la position du visage. Soulignons que les méthodes basées sur le contour les plus abouties reposent sur l'utilisation de contours actifs telles que celles proposées par Waite *et al.* [48], Craw *et al.* [49] et Cootes *et al.* [50]. L'utilisation de contours actifs permet une adaptation aux variations de forme du visage et conduit ainsi à une meilleure délimitation du visage. Une autre approche basée sur les contours consiste en une extraction des contours de l'image et leur mise en adéquation avec une ellipse modélisant un visage [51] [113].

Les méthodes exploitant les traits caractéristiques du visage sont très variées du fait du nombre important de caractéristiques définissables et des techniques de détection. Les traits du visage les plus souvent retenus sont les yeux, les sourcils, la bouche et le nez, qui peuvent être d'un niveau de détail beaucoup plus fin. La détection est alors basée sur le fait que les caractéristiques d'un visage ont des positions relatives fixes (ou statistiquement modélisable).

L'une des premières réalisations est celle de Kanade en 1973 [52] [53]. Cette méthode dédiée à la reconnaissance de visage, constitue un premier pas vers les techniques de détection et de reconnaissance de visage. Les contours de l'image sont tout d'abord extraits puis projetés selon des axes horizontaux et verticaux. Selon le modèle obtenu, le système est capable de retrouver le positionnement des yeux, de la bouche, du nez, ainsi que les contours du visage. Si le profil obtenu ne correspond pas au modèle de référence, l'image est rejetée comme n'étant pas un visage. La méthode la plus répandue pour extraire ces caractéristiques est de réaliser une corrélation entre l'image et un masque de caractéristiques génériques (Sumi et Otha [54], Zelinsky et Heinzmann [55]).

Une autre méthode utilise des filtres successifs pour l'extraction des paramètres du visage, ce qui constitue une approche généralisée de la méthode précédente (Leung *et al.*[56], Graf *et al.* [57], Burl *et al.* [58]). Cependant, l'ensemble des caractéristiques retenues ne constitue pas un modèle robuste et conduit à un grand nombre de fausses alertes. Toutefois, si les positions relatives des traits du visage détectés sont prises en compte par des techniques de comparaison avec un modèle de déformation, alors la robustesse du système est grandement améliorée [113].

Dans [59], Yow et Cipolla décrivent un système de détection du visage où les yeux, la bouche et le nez sont extraits par un filtrage en quadrature de phase et sont regroupés selon leur probabilité à constituer un visage. Le problème de l'angle de vue est résolu par l'utilisation d'invariants augmentant la robustesse de leur approche.

Les approches basées sur l'extraction de traits du visage sont plus efficaces que les approches basées sur les contours car elles analysent des structures locales de l'image conjointement à des modèles géométriques de visages. De plus, il s'agit d'approches ascendantes permettant d'écarter les fausses alertes à chaque étape de la détection d'un nouveau trait du visage [113].

Utilisant la même approche, mais cette fois ci dans le but de modéliser et de suivre les mains, certains chercheurs fournissent une forme 2D de la main. D'autres tels que [116] préfèrent utiliser un modèle générique 3D, où la main est représentée par un modèle de type volumique articulé (cf. figure 2.5). Celui-ci, conformément au paramétrage MPEG-4 [118], peut être animé suivant les 6 degrés de liberté pour le positionnement global, et 5 degrés de liberté pour chaque doigt. Le modèle générique 3D ne correspond pas en général à la morphologie particulière de la main d'un opérateur. L'ajustement préalable des paramètres de morphologie du modèle est nécessaire pour la robustesse de la procédure de suivi de la main dans des séquences. Cette opération est effectuée à partir d'une seule image de la main ouverte (doigts écartés). La silhouette de la main est obtenue à partir d'un seuillage sur la teinte de la peau et les doigts sont séparés par filtrage morphologique. Les paramètres de la morphologie des doigts (rayon et longueur) et de la paume (longueur et largeur) sont calculés à partir des dimensions des régions correspondantes [117].

Figure 2. 5. *Modèle 3D de la main*

2.3 Approches basées sur le mouvement

L'utilisation de l'information de mouvement peut être un moyen simple pour mettre en œuvre une technique rapide de détection de peau dans une vidéo. Cette catégorie de systèmes suppose généralement que l'arrière plan de la scène vidéo est stationnaire et que les régions contenant la peau tels que le visage ou/et les mains par exemple sont en mouvement. Dans ce cas, ces régions peuvent être détectées par une simple différence entre l'image courante et l'image précédente. Il semble évident que les hypothèses retenues sont trop fortes. C'est pourquoi l'information de mouvement n'est jamais utilisée seule pour la détection. On la trouve utilisée conjointement avec l'information de couleur de peau ou couplée à un système basé sur la reconnaissance de traits caractéristiques.

L'exploitation de l'information de mouvement oriente la détection de peau vers des zones préférentielles en éliminant les zones sans intérêt et permet donc une forte réduction de la complexité de la détection et par conséquent une limitation importante des calculs et des traitements [30]. Cependant, les performances de ces systèmes sont fortement réduites lorsque la scène vidéo contient de nombreux objets en mouvement.

Parmi les techniques envisageables, on trouve la soustraction de l'arrière-plan (SAP), la technique basée sur la différence entre deux images consécutives ou encore la technique de calcul du flot optique.

2.3.1 Soustraction de l'arrière plan (SAP) par modélisation statique

La soustraction de l'arrière-plan (SAP) forme la plus grande famille de méthodes de détection du mouvement. Celles-ci sont par ailleurs assez répandues et ont été utilisées dans de nombreux systèmes ([31], [32], [34], [35], [36] et [37]). Ceci s'explique probablement par leur simplicité théorique et leur faible complexité algorithmique. Le principe fondamental repose sur une estimation statistique de la scène observée. Le mouvement est détecté en comparant une image test avec le modèle d'arrière-plan calculé auparavant. Certaines hypothèses de base doivent par contre être respectées pour un fonctionnement adéquat de cette méthode. Tout d'abord, la caméra utilisée est fixe et ne doit bouger à aucun moment. Une caméra à l'épaule est un bon exemple de situation où la SAP ne peut pas s'appliquer.

Le modèle statistique calculé lors de la phase d'initialisation est constamment mis à jour, lui permettant ainsi de s'adapter aux changements qui peuvent se produire dans la scène observée. Cette capacité d'adaptation est commune à toutes les techniques de SAP par modélisation statistique et leur confère un atout majeur. Par ailleurs, cette méthode connaît plusieurs implantations différentes qui varient principalement selon le type de capteur utilisé.

Visible (2D) est la première catégorie de méthodes de soustraction d'arrière-plan qui regroupe les techniques basées sur l'utilisation d'images 2D dans le spectre visible. La technique de base consiste à modéliser l'arrière-plan à partir de plusieurs images acquises séquentiellement. Pour chaque pixel de l'image, ainsi que pour chacun des canaux (R, G et B), une moyenne et une variance sont calculées. Lorsqu'un pixel test doit être classifié, il faut tout d'abord lui soustraire la moyenne correspondante dans le modèle statistique. Il sera alors étiqueté comme un pixel contenant du mouvement seulement si la valeur absolue du résultat dépasse un certain multiple de l'écart type correspondant. Par ailleurs, d'autres modèles de couleur ont été utilisés auparavant pour la modélisation statistique, comme par exemple le YUV par Wren *et al.* [38] et le HSV par Cucchiara *et al.* [34].

Horprasert *et al.* [35] ont proposé un nouveau modèle de couleur basé sur le RGB. Leur technique permet la classification des pixels en quatre catégories : appartenant à arrière plan original, illuminé, ombré, et en mouvement. Pour cela, deux mesures sont ajoutées à la méthode de base en RGB, qui sont la distorsion chromatique (α) et la distorsion de luminosité (CD). Les points faibles de cette approche résident surtout dans la somme d'opérations supplémentaires nécessaires pour calculer ces deux mesures ainsi que les seuils associés. En pratique, certaines erreurs de classification peuvent également se produire entraînant, par exemple, l'identification d'un objet en mouvement comme étant de l'ombre [199].

Il y a finalement un très grand nombre de méthodes de SAP par modélisation statistique qui ne sont pas abordées ici [32], notamment pour des raisons de complexité et pour lesquelles les gains en performance sur la technique de base sont relativement négligeables.

2.3.2 Différence entre deux images consécutives

Etant peu complexe, la différence entre deux images consécutives représente une solution très intéressante. Comme son nom l'indique, elle consiste à soustraire une image acquise au temps t_n d'une autre au temps t_n+k, où k est habituellement égal à 1. Ainsi, l'image résultante

sera vide si aucun mouvement ne s'est produit pendant l'intervalle de temps observé car l'intensité et la couleur des pixels seront presque identiques. Par contre, si du mouvement a lieu dans le champ de vue, les pixels frontières des objets en déplacement devraient changer drastiquement de valeurs, révélant alors la présence d'activité dans la scène. Cette technique nécessite très peu de ressources, car aucun modèle n'est nécessaire. Cela implique donc qu'il n'y a pas de phase d'initialisation obligatoire avec une scène statique, ce qui procure une très grande flexibilité d'utilisation. De plus, une opération de soustraction d'images requiert très peu de puissance de calcul, lui conférant un avantage supplémentaire [44].

Cependant, les résultats obtenus avec cette méthode ne sont pas aussi intéressants que ceux générés en utilisant un modèle statistique de l'arrière plan. En effet, certains traitements supplémentaires sont nécessaires afin de déterminer la zone en mouvement, car l'information disponible ne concerne que les contours des régions en déplacement (ce qui inclus également les zones intérieures d'un objet).

2.3.3 Calcul du flot optique

Les méthodes de calcul de flot optique sont basées sur deux hypothèses: d'une part la photométrie constante d'un pixel et d'autre part le faible espace temporel entre les images sur lesquelles sont calculées le flot optique. En prenant des images très proches les unes des autres, (dans une séquence vidéo par exemple, les images sont prises entre 1/10 et 1/25 seconde d'écart), le mouvement est faible et les pixels en mouvement gardent presque les mêmes valeurs d'intensité.

Le principe consiste à résoudre, pour un pixel ou un bloc, l'équation du flot optique, appelée aussi l'équation de constance de luminance ou l'équation de contrainte du mouvement apparent (ECMA). Elle est la base de toutes les méthodes de flot optique.

$$uI_x + vI_y + I_t = 0 \qquad (2.\ 1)$$

Avec :
$$u = \frac{\Delta x}{\Delta t},\ v = \frac{\Delta y}{\Delta t},\ I_x = \frac{\Delta I}{\Delta x} \qquad (2.\ 2)$$

Afin de résoudre cette équation, il faut en étudier chacun des termes :
- I_x et I_y qui sont obtenus par des gradients spatiaux.
- I_t est un gradient temporel.

Il reste alors 2 inconnues dans l'équation : u et v (supposées indépendantes) qui composent le vecteur de déplacement.

2.4 Approches basées sur la couleur

L'exploitation de l'information de couleur est également un moyen simple et efficace pour la discrimination entre les pixels de peau et ceux de non-peau. De nombreux chercheurs tels que, Fleck *et al.* [92] [93], Kjeldsen *et al.* [63], ont montré que la couleur de la peau est

localisée dans une bande étroite de l'espace de couleur. Cette information peut donc facilement être utilisée pour marquer les pixels de couleur peau. .

Plusieurs modèles de couleur de peau ont été proposés pour la détection des pixels de peau humains dans les images couleur. La majorité de ces modèles utilisent une méthode de segmentation basée sur le calcul d'histogramme de couleur ou une méthode qui dérive de cette dernière [77].

On distingue principalement deux axes de recherche dans cette approche couleur. Le premier, concerne la modélisation de peau/non peau permettant de distinguer les pixels de couleur de peau de ceux de non-peau. Le deuxième axe concerne le choix de l'espace de couleur que nous devons adopter pour une bonne classification.

Etudions d'abord la première direction de recherche sur la construction d'un modèle de peau. Après avoir effectué une étude sur les différents modèles de peau basés sur la couleur, nous avons classifié les travaux en deux sous axes principaux: modèle de peau paramétrique et modèle de peau non paramétrique. Ces derniers font appel respectivement à deux approches de la reconnaissance statistique : (1) les techniques de classification paramétriques, et (2) les techniques de classification non paramétriques. Nous avons jugé utile de présenter dans la section suivante les fondements théoriques des techniques de classification utilisées par de nombreux chercheurs afin d'établir un modèle de peau. Ceci facilitera la compréhension de la majorité des modèles de peau évoqués dans les sections 2.4.2 et 2.4.3.

2.4.1 Fondements théoriques

Les méthodes de classification se déclinent généralement en 2 familles : le mode supervisé et le mode non supervisé. Si l'on dispose d'un ensemble de points étiquetés pour l'apprentissage, on parlera de classification supervisée. Dans le cas contraire, nous parlons d'une classification non supervisée ou classification automatique.

Dans notre cas, les classes sont connues (peau/non-peau), ainsi les méthodes de classification supervisée sont applicables. Ces dernières supposent la connaissance *a priori* de l'appartenance de chaque échantillon de l'ensemble d'apprentissage à une classe donnée, ce qui revient à supposer une connaissance *a priori* sur l'image à segmenter. En partant des données expérimentales que forment les échantillons tirés des classes, on obtient ainsi un effet d'apprentissage : à partir des échantillons soumis au système, ce dernier s'organise en vue de discriminer les échantillons ultérieurs. On dit que le système est capable de généraliser à partir des échantillons d'apprentissage.

On peut distinguer également deux catégories de méthodes de classification : (1) les méthodes indirectes qui utilisent la formule de Bayes, et (2) les méthodes directes qui évaluent les probabilités *a posteriori* sans faire intervenir la formule de Bayes. La formule de Bayes permet de déterminer les probabilités d'appartenance *a posteriori* si les densités de probabilité et les probabilités a priori sont connues. La règle de Bayes permet d'obtenir le taux d'erreur de classification minimum, ce qui est l'objectif souhaitable pour tout système de classification.

Les méthodes indirectes représentent la base de la majorité des travaux de recherche qui traitent la modélisation de la peau. A l'intérieur de ce groupe de méthodes, on distingue encore les méthodes paramétriques (qui font usage d'une hypothèse sur la forme analytique de la distribution) et les méthodes non paramétriques (qui ne font usage d'aucune hypothèse sur la forme de distribution).

Pour résumer, la conception d'un système de classification demande :

1. La spécification des classes $\{Classe_k\}$.
2. La sélection des caractéristiques $\vec{X}$ des classes.
3. La spécification d'une représentation (modèle) pour $P(\vec{X} / Classe_k)$ et $p(Classe_k)$. Cette spécification peut être une fonction paramétrique ou non-paramétrique.
4. L'estimation des paramètres $P(\vec{X} / Classe_k)$ à partir d'un ensemble $S_k = \{ \vec{X}_m \}$ de M_k observations.

Dans notre cas, pour chaque observation $\vec{X}$, la classe $Classe_k$ est connue, et peut être la classe peau ou la classe non-peau. Il suffit donc d'estimer les densités de probabilité.

2.4.1.1 *Règle de décision de Bayes*

Dans la mesure où les vecteurs de caractéristiques sont à N dimensions, les méthodes de la géométrie des espaces sont généralement utilisées. La classification se résume donc en une division de l'espace de caractéristiques en partitions disjointes. Cette division peut être faite par estimation de fonctions paramétriques ou par une liste exhaustive des frontières. Le critère généralement utilisé est la probabilité. Cette probabilité est fournie par la règle de Bayes.

Nous rappelons l'expression mathématique de la formule de Bayes qui prend en considération la probabilité a priori d'apparition des individus des différentes classes et de leur distribution dans l'espace des descripteurs :

$$P(Classe_k / \vec{X}) = \frac{P(\vec{X} / Classe_k)\, p(Classe_k)}{p(\vec{X})} \qquad (2.3)$$

avec $P(Classe_k / \vec{X})$: probabilité *a posteriori* qu'un pixel de caractéristiques $\vec{X}$ appartienne à la classe k.

$P(\vec{X} / Classe_k)$: densité de probabilité de $\vec{X}$ si la classe est k.

$p(Classe_k)$: probabilité *a priori* qu'un pixel appartienne à la classe k.

$P(\vec{X})$: probabilité d'observer $\vec{X}$.

La règle de décision de Bayes consiste à affecter l'individu à la classe dont la probabilité a posteriori (calculé par la formule de Bayes ou par toute autre méthode) est la plus grande. Cette décision minimise la probabilité d'erreur de classement. La figure 2.6 apporte une

explication géométrique. Duda et Hart [115] donnent une preuve mathématique plus rigoureuse.

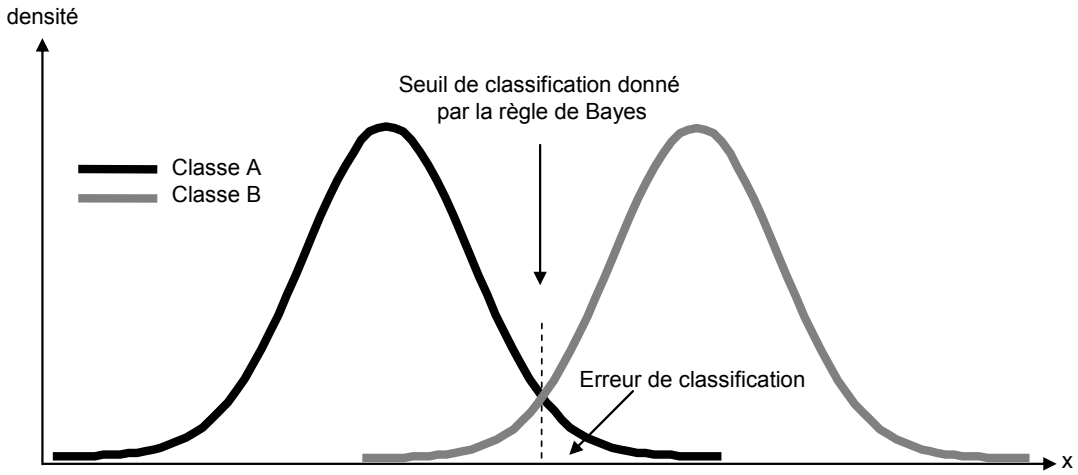


Figure 2. 6. *Distribution des individus dans l'espace de description*

Les courbes représentent les fonctions densités de probabilité pondérées par les probabilités a priori correspondant aux deux classe A et B. De cette manière, ces courbes sont directement reliées à la densité des individus. Le trait vertical marque le seuil de classification donné par la règle de Bayes entre les deux classes.

L'erreur de classification est le nombre d'exemples A classée comme B et inversement. Elle correspond donc à la surface d'intersection des deux courbes. Si l'on choisit un autre seuil, nous nous apercevons que la nouvelle erreur de classification est égale à l'erreur de classification de Bayes augmentée d'une contribution positive. Elle est donc toujours supérieure à l'erreur de Bayes. Ainsi quel que soit le seuil pris pour séparer les 2 classes, l'erreur de classification est toujours supérieure à celle trouvée avec la règle de Bayes.

2.4.1.2 *Estimation paramétrique des densités de probabilité*

Les méthodes paramétriques consistent à faire une hypothèse concernant la forme analytique de la distribution de probabilité recherchée, et à estimer les paramètres de cette distribution à partir des données dont on dispose. En d'autres termes, à l'aide de quelques paramètres (moyenne, variance…), on ajuste la loi de distribution choisie par rapport aux individus à notre disposition. On obtient une estimation des paramètres, et l'on peut ensuite utiliser la forme analytique de la densité ainsi déterminée pour en déduire la densité en tout point de l'espace de représentation.

L'hypothèse la plus courante est que la répartition des individus de chacune des classes suit une loi gaussienne (figure 2.7). Elle conduit à la méthode appelée analyse discriminante avec une règle d'affectation probabiliste. Cette distribution « normale » des individus est la plus utilisée.

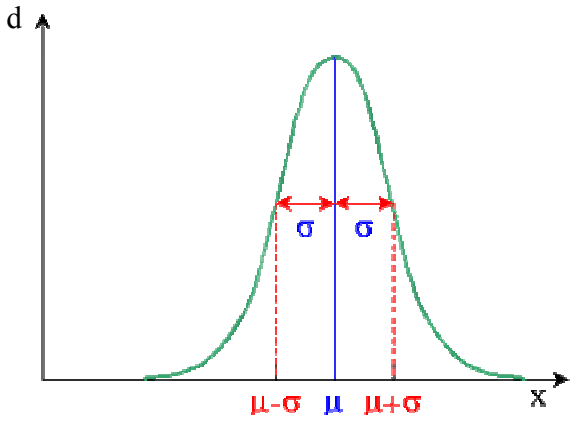


Figure 2. 7. *Distribution relative à une loi normale*

μ : moyenne de la gaussienne.
σ : écart type
d : distribution des individus

2.4.1.2.1 Analyse discriminante avec une règle d'affectation probabiliste

On rappelle l'hypothèse de distribution : les individus de la classe k sont répartis suivant une loi gaussienne multidimensionnelle :

$$P(x/Classe_k) = N(\vec{\mu}, \Sigma) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)} \quad (2.\ 4)$$

avec Σ_k : matrice de covariance de la classe k,
μ_k : moyenne de la gaussienne de la classe k.

La matrice de covariance et le centre de la classe k sont estimés par la matrice de covariance et la moyenne des individus appartenant à la classe k.

Ainsi, à partir des estimations des matrices de covariance, des moyennes des gaussiennes (pour chacune des classes) et des probabilités *a priori*, on calcule par la formule de Bayes les probabilités *a posteriori* d'appartenance d'un individu aux classes. La règle de décision consiste à classifier un individu dans la classe qui obtient la plus grande probabilité *a posteriori*. La frontière de séparation est donc déterminée par l'ensemble des points pour lesquels les probabilités *a posteriori* sont égales.

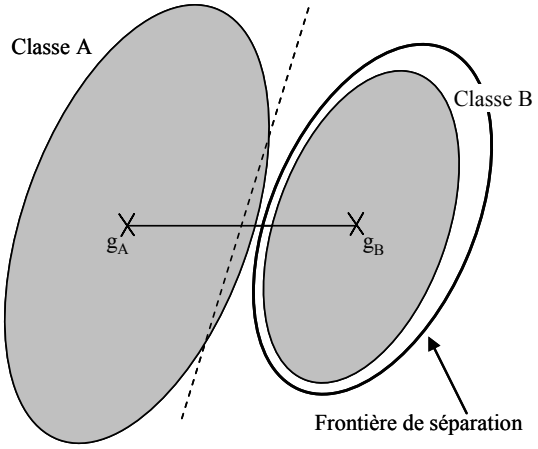


Figure 2. 8. *Frontière de séparation*
(Analyse discriminante avec une règle d'affectation probabiliste)

Sur la figure 2.8, on remarque que la frontière de séparation prend en considération la différence des classes. Le trait pointillé matérialise la frontière obtenue avec la règle d'affectation géométrique (voir section suivante).

Cette forme d'analyse discriminante peut sembler a priori très intéressante. Malheureusement même si les hypothèses sont vérifiées, les estimations des différentes matrices sont effectuées à partir des exemples qui risquent d'être peu nombreux. Elles sont donc très sensibles aux exemples marginaux.

De façon pratique, on préfère ajouter quelques hypothèses qui conduisent à l'analyse discriminante avec une règle d'affectation géométrique.

2.4.1.2.2 Analyse discriminante avec une règle d'affectation géométrique

C'est la forme la plus simple de l'analyse discriminante. L'hypothèse de départ est complétée par les hypothèses suivantes pour garantir une convergence vers la règle de Bayes :

- Les individus de la classe k sont répartis suivant une loi gaussienne multidimensionnelle déterminée par :

$$P(x/Classe_k) = N(\vec{\mu}, \Sigma) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)} \qquad (2.\ 5)$$

avec Σ_k : matrice de covariance de la classe k,
μ_k : moyenne de la gaussienne de la classe k.

- Les différentes matrices de covariance sont identiques :

$$\Sigma_1 = \Sigma_2 = \cdots = \Sigma_C = \Sigma \qquad (2.\ 6)$$

- Les probabilités a priori des classes sont, elles aussi, identiques :

$$Pr_1 = Pr_2 = \cdots Pr_C = 1/C \qquad (2.\ 7)$$

Dans ce cas, pour classer un nouvel exemple, l'analyse discriminante avec affectation géométrique calcule la distance (métrique de Mahalanobis) entre l'exemple et les différents centres de gravités des classes, et affecte à cet exemple la classe correspondant à la plus petite distance. La distance de Mahalanobis (notée Δ_Σ) est donc définie globalement dans l'espace de description par la matrice de covariance des individus :

$$\Delta^2_\Sigma(\mu_1, \mu_2) = (\mu_1 - \mu_2)^T \Sigma^{-1}(\mu_1 - \mu_2) \qquad (2.\ 8)$$

avec μ_1 et μ_2 : 2 vecteurs dans l'espace de description.

Cette méthode est dite *géométrique* car elle ne tient compte que de l'éloignement de l'exemple considéré aux centres de gravité : elle revient à découper l'espace par les hyperplans médiateurs des segments joignant les centres de gravité (au sens de la métrique utilisée).

Dans le cas de la classification à deux classes, on introduit la fonction discriminante de Fisher [119] qui est donnée par :

$$w = e^T \Sigma^{-1}(\mu_1 - \mu_2) \qquad (2.\ 9)$$

où w est la valeur de la fonction discriminante de Fisher au point de coordonnées e et μ_k est la moyenne de la gaussienne de la classe k.

Ainsi, on affectera l'observation e à la classe 1 si :

$$w = e^T \Sigma^{-1}(\mu_1 - \mu_2) > \frac{1}{2}(\mu_1 - \mu_2)^T \Sigma^{-1}(\mu_1 - \mu_2) \qquad (2.\ 10)$$

La figure 2.9 montre un exemple de classification à deux classes :

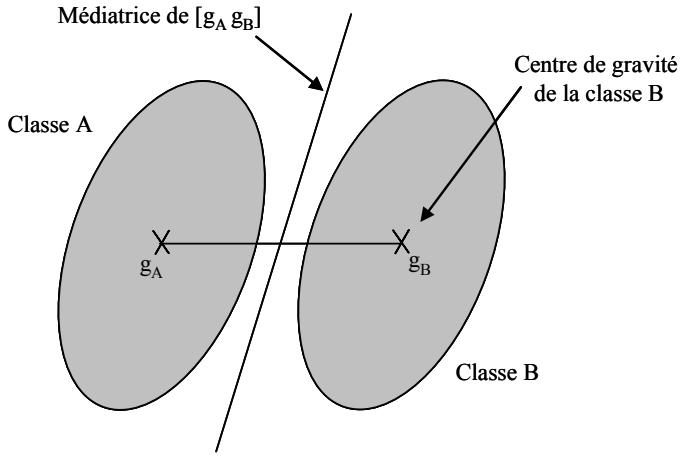


Figure 2. 9. *Frontière de séparation*
(Analyse discriminante avec une règle d'affectation géométrique)

Au sens de la métrique de Mahalanobis, la frontière entre les deux classes (A et B) est bien la médiatrice du segment [$g_A g_B$].

Cette méthode de classification est très simple à mettre en œuvre, car elle sépare les classes suivants des hyperplans (fonctions linéaires), malheureusement le résultat obtenu est rarement (voire jamais) celui que l'on obtiendrait par le classifieur de Bayes. Ainsi, une configuration typique est celle de la figure suivante :

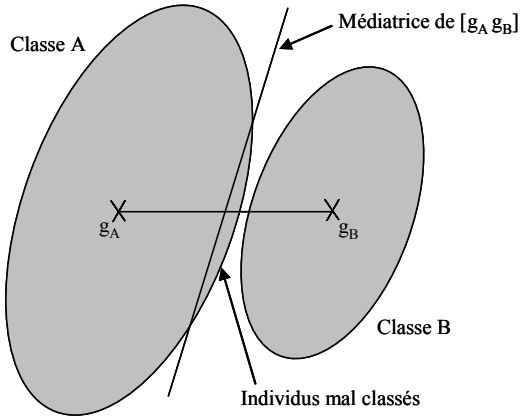


Figure 2. 10. *Frontière de séparation*
(analyse discriminante avec une règle d'affectation géométrique)

Ici, les individus de la classe A sont plus dispersés que ceux de la classe B. La frontière, quant à elle, n'a pas bougé par rapport à la figure 2.9 puisque les centres de gravité sont restés identiques. De nombreux individus de la classe A sont donc mal classés.

2.4.1.2.3 Estimation de la moyenne (μ) et la covariance (∑)

Nous rappelons que pour représenter $P(\vec{X} / Classe_k)$, les densités de probabilité de $\vec{X}$ peuvent êtres estimées par une loi normale.

$$P(\vec{X} / Classe_k) = N(\vec{\mu}, \Sigma) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}} e^{-\frac{1}{2}(\vec{X}-\vec{\mu})^T \Sigma_k^{-1}(\vec{X}-\vec{\mu})} \quad (2.\ 11)$$

L'apprentissage dans ce cas est réduit au problème d'estimation de la moyenne et de la covariance.

a. Estimation de la moyenne (μ)

Soit M observations d'une variable aléatoire, $\{x_i\}$
La moyenne, μ_i, est l'espérance de $\{x_i\}$. Elle est donnée par la formule (2.12)

$$\mu \equiv E\{x\} = \sum x\ p(x). \quad (2.\ 12)$$

Pour les vecteurs de propriétés :

$$\vec{\mu} \equiv E\{\vec{X}\} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \dots \\ \mu_n \end{pmatrix} = \begin{pmatrix} E\{x_1\} \\ E\{x_2\} \\ \dots \\ .E\{x_n\} \end{pmatrix} \quad (2.\ 13)$$

Chaque composante est

$$\mu_i \equiv E\{x_i\} = \sum x_i\ p(x_i). \quad (2.\ 14)$$

Cette moyenne peut être estimée par l'histogramme. On définit *M* la masse d'un histogramme h(x) comme le nombre d'échantillons qui le composent.

$$M = \sum_{x_{\min}}^{x_{\max}} h(x) \quad (2.\ 15)$$

On obtient ainsi :

$$\mu \equiv E\{x\} = \sum_{x_{\min}}^{x_{\max}} p(x).x \approx \frac{1}{M} \sum_{x_{\min}}^{x_{\max}} h(x)\, x \quad (2.\ 16)$$

Pour un vecteur de variables à *n* dimensions, on aura :

$$\vec{\mu} \equiv E\{x_n\} = \sum_{x_{\min}}^{x_{\max}} p(\vec{X}).x_n \approx \frac{1}{M}\sum_{x_{\min}}^{x_{\max}} h(\vec{X}).x_n \qquad (2.\ 17)$$

b. Le deuxième moment : la covariance

La variance σ^2 est le deuxième moment de la densité de probabilité.
Pour un ensemble d'observations *M*, la variance de la variable x_i est :

$$\sigma^2 \equiv E\{(x-\mu)^2\} = \frac{1}{M}\sum_{i=1}^{M}(x_i-\mu)^2 \qquad (2.\ 18)$$

Mais l'usage de μ estimé avec le même ensemble, introduit un biais dans σ^2. Pour l'éviter, on utilise une estimation sans biais.

$$\sigma^2 = \frac{1}{M-1}\sum_{i=1}^{M}(x_i-\mu)^2 \qquad (2.\ 19)$$

Avec μ et σ^2 on peut estimer la densité p(x) par :

$$P(x) = N(\vec{\mu},\sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad (2.\ 20)$$

σ^2 est le deuxième moment de p(x).

$$\sigma^2 \equiv E\{(x-\mu)^2\} = \sum_{x=x_{\min}}^{x_{\max}} p(x).(x-\mu)^2 = \frac{1}{M}\sum_{x=x_{\min}}^{x_{\max}} h(x)(x-\mu)^2 \qquad (2.\ 21)$$

Pour *N* dimensions, la covariance entre les variables x_i et x_j est estimée à partir de M observations

$$\sigma_{ij}^2 \equiv E\{(x_i - E\{x_i\})(x_j - E\{x_j\})\} \qquad (2.\ 22)$$

$$= \frac{1}{M}\sum_{m=1}^{M}(x_{im}-\mu_i)(x_{jm}-\mu_j) \qquad (2.\ 23)$$

et encore pour éviter le biais, on utilise :

$$= \frac{1}{M-1}\sum_{m=1}^{M}\left(x_{im}-\mu_i\right)\left(x_{jm}-\mu_j\right) \qquad (2.\ 24)$$

Ces coefficients composent la matrice de covariance Σ

$$\Sigma_x \equiv E\left\{\left(\vec{X}-\vec{\mu}\right)\left(\vec{X}-\vec{\mu}\right)^T\right\} = E\left\{\left(\vec{X}-E\left\{\vec{X}\right\}\right)\left(\vec{X}-E\left\{\vec{X}\right\}\right)^T\right\} \qquad (2.\ 25)$$

$$\Sigma_x \equiv \begin{pmatrix} {\sigma_{11}}^2 & {\sigma_{12}}^2 & \dots & {\sigma_{1n}}^2 \\ {\sigma_{21}}^2 & {\sigma_{22}}^2 & \dots & {\sigma_{2n}}^2 \\ \dots & \dots & & \dots \\ {\sigma_{n1}}^2 & {\sigma_{n2}}^2 & \dots & {\sigma_{nn}}^2 \end{pmatrix} \qquad (2.\ 26)$$

On obtient les paramètres de la fonction

$$P(\vec{X}/Classe_k) = N(\vec{\mu},\Sigma) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}} e^{-\frac{1}{2}(\vec{X}-\vec{\mu})^T\sum_k^{-1}(\vec{X}-\vec{\mu})} \qquad (2.\ 27)$$

2.4.1.3 *Estimation non paramétrique des densités de probabilité*

On a traité jusqu'ici la construction d'un classificateur sous l'hypothèse d'une loi de distribution explicitement paramétrée, pour chaque classe. Malheureusement la forme de cette distribution est souvent difficile à choisir par manque d'un modèle suffisamment adéquat et complet de la réalité. Lorsqu'on ne peut pas faire d'hypothèse sur la distribution des individus, il faut se tourner vers des méthodes non paramétriques. Il s'agit de s'affranchir de l'hypothèse d'une loi paramétrique et d'estimer la distribution $P(x)= P(X=x\backslash K=i)$ de la classe courante par approximations.

Le principe de l'estimation non paramétrique de la densité de probabilité est de délimiter une région $\boldsymbol{R}_N$ autour d'un point considéré, puis de compter le nombre d'individus dans ce volume, et enfin de déterminer la densité comme le rapport entre ce nombre (divisé par le nombre total d'individus) et le volume de la région ([74], [75] et [76]). Ainsi, on obtient une estimation de la densité de probabilité avec la formule suivante :

$$\hat{p}_N(x) = \frac{k_N}{N.V_N} \qquad (2.\ 28)$$

Avec :

N : nombre d'individus de l'échantillon
K_N : nombre d'individus dans la région $\boldsymbol{R}_N$
V_N : volume de $\boldsymbol{R}_N$

Ayant une telle fonction, on peut utiliser une technique optimale de classification en appliquant la règle de Bayes.

Certaines techniques non-paramétriques placent une surface de décision autour de chaque échantillon d'une classe. D'autres estiment la densité de probabilité par une table de fréquence d'occurrence (un histogramme).

Le problème principal dans l'estimation d'une fonction de densité pour un vecteur de caractéristiques est la croissance exponentielle du nombre de cellules *N* avec le nombre de dimensions *D*. Ceci induit une croissance exponentielle dans les nombres d'exemples *M* nécessaires. En Anglais, on appelle ce problème "The Curse of Dimensionality".

2.4.2 Modèle de peau paramétrique

Les modèles de peau paramétriques permettent d'ajuster les distributions avec quelques fonctions spécifiques paramétrées. Ils offrent trois avantages : (1) ils apportent un gain en espace mémoire ainsi qu'en possibilité de manipulation ; (2) ils donnent plus d'intelligence et de finesse dans les vraies formes ou dans la régularité des distributions ; et (3) ils ont la capacité d'interpoler les données d'apprentissage quand elles sont dispersées. Différentes fonctions peuvent être appliquées en fonction du problème. Nous présentons dans la suite les méthodes paramétriques les plus utilisés pour construire un modèle de peau et de non peau.

2.4.2.1 *Modèle basé sur une simple gaussienne*

Cette méthode a été utilisée par [68], [69] et [72]. La distribution de couleur de peau est estimée par une fonction de densité de probabilité gaussienne :

$$p(c|peau) = \frac{1}{2\pi\sqrt{|\Sigma_{peau}|}} e^{-\frac{1}{2}(c-\mu_{peau})^T \Sigma_{peau}^{-1}(c-\mu_{peau})} \qquad (2.\ 29)$$

où c est la variable aléatoire à deux dimensions représentant le couple de chrominance; μ_{peau} et Σ_{peau} sont respectivement l'espérance et la matrice de covariance représentant les paramètres du modèle gaussien. Ces paramètres peuvent être estimés à partir de l'échantillon d'apprentissage selon les équations suivantes :

$$\mu_{peau} = \frac{1}{N_{peau}} \sum_{c \in C} N_{peau}(c)\, c \qquad (2.\ 30)$$

$$\Sigma_{peau} = \frac{1}{N_{peau} - 1} \sum_{c \in C} N_{peau}(c)(c - \mu_{peau})(c - \mu_{peau})^T \qquad (2.\ 31)$$

2.4.2.2 *Modèle basé sur un mélange de Gaussiennes.*

D'autres auteurs [70], [71], [73], [18], [66] et [67] ont également proposé l'utilisation d'un modèle mixte de gaussiennes (Gaussian Mixture Model), qui vise à mieux représenter et modéliser la portion d'un espace de couleur associée à la couleur peau .

En général les densités de probabilités sont plus complexes que $N(\mu,\sum)$. Une fonction paramétrique plus générale est un mélange de Gaussiennes.

Le mélange des Gaussiennes est une extension des gaussiennes simples. A la différence d'une gaussienne simple, il a la capacité de représenter les distributions les plus complexes. Dans ce cas la fonction de densité de probabilité gaussienne est représentée comme suit:

$$p(c \mid peau) = \sum_{n=1}^{N} w_n p_n \left(c \middle| peau\right) \qquad (2.\ 32)$$

Avec :
p_n : les noyaux des gaussiens définis dans (2.29). Chaque p_n est lui-même une distribution gaussienne.
N est le nombre de noyaux gaussiens qu'il faut choisir correctement de sorte que le modèle puisse bien représenter les données d'apprentissage
w_n sont les poids des noyaux correspondants dont la somme est égale à 1.

Dans ce cas, trois paramètres sont à estimer (w_n, μ_n, Σ_n). Généralement l'algorithme EM (Expectation Maximization) est appliqué pour estimer ces paramètres.

La figure 2.11 illustre un mélange de trois gaussiennes en proportions égales (à gauche) et l'histogramme correspondant (à droite). La courbe en trait plein à gauche représente la densité de probabilité théorique du mélange résultant en tenant compte des proportions.

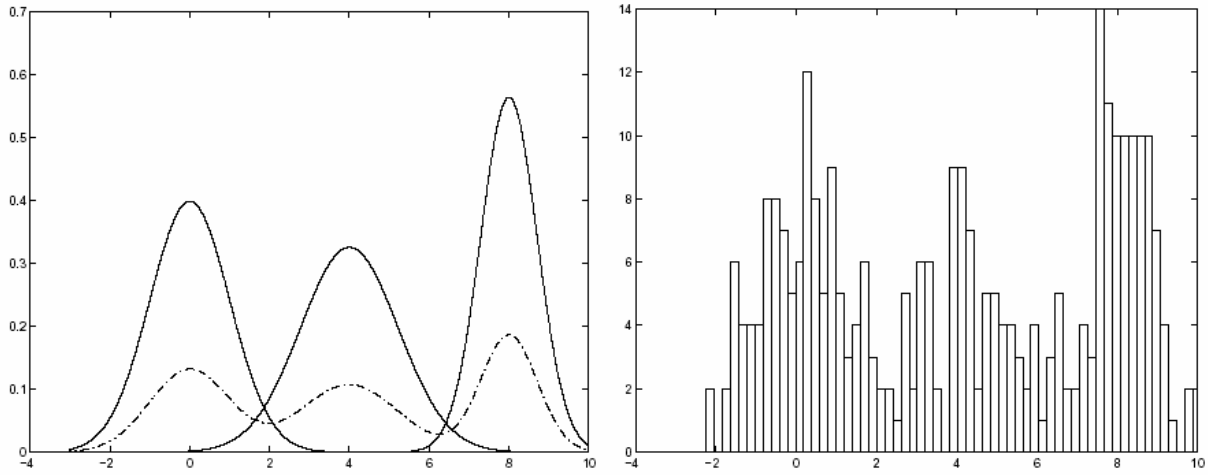


Figure 2. 11. *(A gauche) un mélange de 3 gaussiennes (en pointillé) et leurs densités théoriques (trait plein) (A droite) l'histogramme associé.*

2.4.2.3 *Modèle elliptique de borne*

En examinant la distribution de couleur de peau et de non-peau dans plusieurs espaces de couleur, Lie et Yoo [98] ont conclut que cette distribution prend approximativement la forme d'une ellipse. Ils ont séparé les régions de couleur de peau et de non-peau par une borne elliptique afin de caractériser la vraie forme de la distribution des données d'apprentissage. Le modèle est défini comme suit:

$$\Phi(c) = (c - \phi)^T \sum_{peau}^{-1} (c - \phi) \qquad (2.\ 33)$$

Où ϕ et Σ_{peau} sont calculés à partir des données d'apprentissage comme suit :

$$\phi = \frac{1}{|C_{peau}|} \sum_{c \in C_{peau}} c \qquad (2.\ 34)$$

$$\Sigma_{peau} = \frac{1}{N_{peau}} \sum_{c \in C_{peau}} N_{peau}(c)(c - \mu_{peau})(c - \mu_{peau})^T \qquad (2.\ 35)$$

où $|C_{peau}|$ est le nombre de pixels de couleur de peau avec $C_{peau} \subseteq C$. μ_{peau} est l'espérance des vecteurs d'apprentissage de pixels de peau défini par l'équation (2.30).

Pour classifier les pixels on compare $\Phi(c)$ avec un seuil fixé θ. c est considéré comme un pixel de peau si $\Phi(c) < \theta$ sinon il est un pixel de non peau.

Les taux de classification obtenus par ce modèle sur six espaces de couleur testés sont satisfaisants. De plus, cette méthode est plus rapide lors de la classification que les méthodes basées sur une simple gaussienne [98].

2.4.3 Modèle de peau non paramétrique

La recherche selon cette approche vise à estimer la distribution de couleurs de peau des données d'apprentissage sans aucune hypothèse sur la distribution.

2.4.3.1 *Classification de pixels par table de correspondance : lookup table (LUT)*

Cette méthode consiste à construire dans un premier temps un modèle de peau en se basant sur le calcul d'histogramme à partir d'exemples. Elle l'appliquer ensuite à une image ou à une scène pour obtenir une carte de probabilité en associant à chaque pixel sa probabilité d'appartenance à une instance du modèle. A partir de cette carte, les pixels sont classés. Cette méthode a été appliquée dans plusieurs algorithmes de détection et de suivi de visage pour segmenter les pixels de peau ([82], [64], [79],[110], [111], [104] et [89]).

Après projection dans un espace de couleur bien choisi, le modèle prend la forme d'un histogramme des valeurs des pixels de la base d'exemples. En effet, l'espace de couleur est quantifié par un nombre de case (bins). A chaque case des axes utilisés est associé le nombre de fois que la valeur de couleur s'est produite dans les images de peau de la base d'apprentissage. Ces cases forment donc un histogramme de couleur (3D) ou (2D) désigné dans certains travaux sous le nom de table de correspondances « Lookup Table ».

Nous signalons que les histogrammes (2D) ont été plus utilisés dans la littérature. Nous citons l'espace (*r*, g) (composantes *R* et *G* normalisées) par Bérard [28], l'espace (*H*, *S*) (teinte et saturation) par Wu [39]. En effet, à partir d'un histogramme tri-dimensionnel (3-D) on peut déduire trois histogrammes (2-D), par projection de l'histogramme (3-D) sur 2 des 3 plans colorimétriques d'une image *I*. Un histogramme (2-D) peut donc être considéré comme une image *J* dont les dimensions spatiales sont celles des deux axes choisis dans l'espace colorimétrique associé à *I*.
D'autres chercheurs préfèrent l'utilisation d'un histogramme tridimensionnel. Quek *et al.* [100] ont construit leur table de correspondances dans l'espace RGB à partir d'une base d'images prise contre un arrière plan simple afin de permettre une segmentation facile des pixels.

Une fois l'histogramme construit, il est normalisé et ces valeurs sont converties en distribution discrète $p(c|peau)$.

$$p(c|peau) = \frac{peau(c)}{Norm} \qquad (2.\ 36)$$

où peau(c) est le nombre de pixels associé à une case (bins) de l'histogramme de peau formé par le vecteur de couleur *c* et *Norm* est le coefficient de normalisation.

Ce coefficient (*Norm*) a été fixé selon différentes manières. Il correspond au nombre total de pixels de peau dans les travaux de [112] et [100]. Il est égal à la plus grande valeur dans la table de correspondance (Lookup Table) dans les travaux de [82]. Les valeurs normalisées de cette table constituent la probabilité qu'une couleur correspond à la peau. Un pixel est considéré comme pixel de peau si sa valeur dépasse un certain seuil.

Swain et Ballard [29] montrent que l'histogramme de chrominance est un modèle fiable pour la reconnaissance d'entités colorées. Ils expérimentent différents types d'histogrammes dont ceux créés à partir des composantes rouges et vertes normalisées. Une cellule de coordonnées (*r*, *g*) de l'histogramme à deux dimensions h_E donne le nombre de pixels de l'échantillon *E* ayant une chrominance de composante rouge *r* et verte *g*. Cet histogramme permet de définir la probabilité de l'équation 2.36 par :

$$P(c \setminus peau) = \frac{1}{n_E} h(c) = \frac{1}{n_E} h(c_r, c_g) \qquad (2.\ 37)$$

Où n_E est le nombre total de pixels de l'ensemble *E*.

Pour classifier les pixels en pixels de peau ou non peau, Cai *et al.* [102] utilisent une table de correspondances construite selon les deux axes de couleur *a* et *b* de l'espace CIE Lab. La table a été établie à partir de 2300 échantillons de peau extraits de 80 images contenant des pixels de peau. Afin de réduire l'effet du bruit dans les échantillons, on calcule à chaque pixel la couleur moyenne dans un voisinage de taille 3 x 3. Après balayage de tous les pixels de la base d'apprentissage, on applique pour chaque valeur d'entrée de la table de chrominance un filtre gaussien. Ce dernier complète les petites zones manquées du diagramme de chrominance. Pour classifier les pixels d'une image, une image de probabilité de peau est établie selon le diagramme de chrominance. De cette image les auteurs localisent les pics locaux, servant comme germes pour effectuer la segmentation de l'image. Le résultat est une image contenant des régions de peau et des régions non-peau. Cette méthode est dédiée à la détection de visage c'est pourquoi les auteurs comparent chaque région de peau avec un template de visage.

2.4.3.2 *Modèle basé sur l'appariement d'histogrammes*

Saxe et Foulds [91] utilisent l'espace de couleur HSV et une technique basée sur l'appariement d'histogrammes, décrite dans [29]. L'histogramme de couleur calcule la fréquence d'apparition de chaque couleur. Cette information est globale et elle ne prend pas en compte la disposition spatiale des pixels dans l'image. Elle est donc invariable à la rotation et à la translation, ce qui est une propriété intéressante pour apparier. Les auteurs suggèrent une sélection d'une région de peau, appelée région de contrôle, qui est comparée ensuite au reste de l'image en utilisant l'appariement d'histogramme. Il s'agit de comparer l'histogramme de la région de contrôle avec les histogrammes des régions de même taille de l'image. Dans un premier temps l'algorithme transforme l'image d'entrée dans l'espace de couleur HSV. Puis c'est à l'utilisateur de choisir manuellement le germe initial (région de contrôle). Une fois le bloc de contrôle choisi, l'image est examinée région par région en utilisant la méthode d'intersection d'histogramme (équation 2.38).

$$M_{C,I} = \frac{\sum_{i,j} \min\left(H^C(i,j), H^I(i,j)\right)}{\sum_{i,j} H^C(i,j)} \qquad (2.38)$$

Les histogrammes sont calculés en se basant sur les valeurs de teinte et de saturation. Si le score d'appariement ($M_{C,I}$) entre l'histogramme de contrôle (H^C) et l'histogramme correspondant à un bloc de l'image (H^I), excède un seuil, le bloc est considéré comme peau. Une fois que tous les blocs de l'image ont été examinés, l'identification de blocs de peau sera améliorée par élimination de blocs annexes. Un bloc est considéré comme annexe quand aucun de ses huit voisins n'a été marqué comme peau. Finalement, l'algorithme choisit un nouveau bloc de contrôle parmi les blocs de peau qui ont été conservés (ne comprenant pas les blocs retirés). Le processus sera répété. Quand aucun nouveau bloc n'a été identifié pendant l'itération en cours, le processus d'identification des blocs de peau est considéré comme achevé. Les auteurs ne fournissent pas beaucoup de détails sur la façon de choisir un nouveau bloc de contrôle. Deux méthodes différentes présentent des voies possibles. La première consiste à choisir comme nouveau bloc celui qui correspond au score d'appariement le plus élevé ($M_{C;i}$), alors que pour la deuxième méthode, le choix du nouveau blocs de contrôle se porte sur celui qui correspond au score le moins élevé.

Toutefois, cet algorithme rencontre plusieurs problèmes. Le premier correspond au choix du bloc initial de contrôle qui est parfois trop critique. En effet dans certains cas, en utilisant un même seuil, on peut avoir deux résultats différents si l'emplacement de la sélection initiale diffère seulement par deux pixels. En effet, cette sélection permet d'avoir deux blocs de contrôle séparés. Ces deux blocs adjacents donnent des résultats très différents. Ainsi cet algorithme exige une bonne sélection initiale. En revanche dans certain cas l'utilisateur peut choisir arbitrairement la meilleure sélection initiale.

Le second problème réside dans le choix du seuil adéquat. En effet, le seuil est choisi manuellement. Un petit changement du seuil a comme conséquence un grand changement des résultats. Utilisant un premier seuil X, un nombre limité de bloc serait identifié comme des blocs de peau. Un deuxième seuil X-0.01 permet d'identifier un nombre important de blocs de peau.

A notre connaissance, la cause de ces deux problèmes n'a été jamais déterminée. Toutefois, ces problèmes peuvent être liés au choix des nouveaux blocs. Changer le seuil peut changer le bloc utilisé dans chaque itération. Pour conclure cet algorithme a très bien fonctionné dans certains cas, alors qu'il a mal fonctionné dans beaucoup d'autres cas.

Ahmad [101] utilise également une méthode basée sur l'appariement des histogrammes pour segmenter les régions de peau dans l'image.

2.4.3.3 *Modèle Bayésien basé sur les Histogrammes*

Dans [80] [81], les auteurs présentent leur modèle de couleur de peau et de non-peau par des histogrammes. Ils quantifient l'espace de couleur C à un certain nombre de bins $c \in C$ et comptent le nombre de pixel de couleur dans chaque bins. $N_{peau}(c)$ représente donc le nombre de pixels de couleur c pour la classe de peau et $N_{\neg peau}(c)$ pour la classe de non-peau. Enfin ils normalisent chaque bin pour obtenir la distribution conditionnelle des pixels de peau et de non peau $p(c|peau)$ et $p(c|non\ peau)$. Supposons que N_{peau} représente le nombre total de pixel de peau et $N_{\neg peau}$ le nombre total de pixels de non-peau dans la base d'apprentissage, nous avons :

$$p(c|peau) = \frac{N_{peau}(c)}{N_{peau}} \tag{2. 39}$$

$$p(c|\neg peau) = \frac{N_{\neg peau}(c)}{N_{\neg peau}} \tag{2. 40}$$

Ce qui signifie aussi :

$$p(peau) = \frac{N_{peau}}{N_{peau} + N_{\neg peau}} \tag{2. 41}$$

$$p(\neg peau) = \frac{N_{\neg peau}}{N_{peau} + N_{\neg peau}} = 1 - p(peau) \qquad (2.\ 42)$$

Par la suite, la formule de Bayés ci dessous est utilisée pour calculer la probabilité qu'un pixel soit un pixel de peau ou non selon sa couleur.

$$p(peau|c) = \frac{p(c|peau)\,p(peau)}{p(c|peau)\,p(peau) + p(c|\neg peau)\,p(\neg peau)} \qquad (2.\ 43)$$

$$p(\neg peau|c) = 1 - p(peau|c) \qquad (2.\ 44)$$

La décision concernant les deux classes est faite selon un seuil choisi Θ, $0 < \Theta < 1$. Le pixel sera considéré comme un pixel de peau si $p(peau|c) > \Theta$ et comme un pixel de non-peau si $p(peau|c) <= \Theta$.

2.4.3.4 *Self-organizing map (SOM)*

L'algorithme SOM (Self-Organizing Map)[120], encore appelé algorithme de Kohonen, est l'un des plus populaire et des plus utilisés des réseaux neuronaux artificiels non supervisés[33].

Brown *et al.*[83] ont appliqué l'algorithme (SOM) pour identifier les pixels de peau dans l'image, comme étape préliminaire dans leur système de détection de visages. Ils ont construit deux SOMs, le premier ne représente que des pixels de peau, appris à partir de 30000 pixels de peau, alors que le deuxième représente des pixels de peau et de non peau où 15000 pixels de chaque classe ont été utilisés.

Le principe de la carte auto adaptative de Kohonen est le suivant :
Soit $\Gamma = \{X_1, X_2, \ldots, X_q, \ldots, X_Q\}$ un échantillon constitué de Q observations définies dans un espace à N dimensions telles que $X_q = [x_{q,1}, x_{q,2}, \ldots, x_{q,n}, \ldots, x_{q,N}]^T$, $q=1, 2, \ldots, Q$. La structure du réseau de Kohonen est constituée de deux couches. La première, appelée couche d'entrée, est composée de N neurones représentants les N attributs d'une observation X_q. La couche de sortie, ou couche compétitive, est composée d'un nombre M de neurones régulièrement répartis sur une carte.

Les neurones de la première couche sont connectés à ceux de la deuxième couche. Chaque connexion d'un neurone d'entrée j vers un neurone de sortie m a le vecteur poids $W_{m,j}$. Ainsi chaque neurone de sortie m a le vecteur poids :

$$W_m = [w_{m,1}, w_{m,2}, \ldots, w_{m,n}, \ldots, w_{m,N}]^T \qquad (2.\ 45)$$

Chaque neurone de la couche de sortie est donc caractérisé par sa position relative sur la carte et par son vecteur poids dans l'espace de représentation des observations. L'apprentissage du réseau consiste, à chaque présentation à l'itération *t* d'une observation $X_q(t)$ à l'entrée du réseau, à sélectionner le neurone gagnant, autrement dit celui dont le vecteur poids est le plus proche de cette observation. Le vecteur poids du neurone gagnant et ceux des neurones voisins sur la carte sont alors modifiés en fonction de l'observation présentée au réseau. Cette disposition des neurones et cette technique d'apprentissage permettent aux neurones voisins sur la carte d'être sensibles à des observations voisines dans l'espace d'origine : c'est le phénomène d'auto-adaptation décrit par Kohonen [121]. A la fin de cette étape, chaque neurone devient sensible à une zone de l'espace de représentation des observations et son vecteur poids converge vers le barycentre des observations présentes dans cette zone.

Le voisinage de rayon *r* d'un neurone *m* est défini par l'ensemble des neurones *m'* tel que :

$$V(m,r) = \{m' \in [0,M[, m' \neq m / d(U_m, U_{m'}) \leq r\} \qquad (2.\ 46)$$

où $d(U_m, U_{m'})$ désigne la distance Euclidienne entre les vecteurs U_m et $U_{m'}$ qui sont les positions respectives des neurones *m* et *m'* sur la carte.

A la présentation au réseau de chaque observation $X_q(t)$, le vecteur poids du neurone gagnant, noté m^*, et ses voisins sont modifiés pour chaque itération *t* tels que :

$$W_m(t) = W_m(t-1) + a(t).[X_q(t) - W_m(t-1)] \quad \text{si } m=m^* \qquad (2.\ 47)$$

$$W_m(t) = W_m(t-1) + a(t).[X_q(t) - W_m(t-1)] \quad \text{si } m=m^* \qquad (2.\ 48)$$

$$W_m(t) = W_m(t-1) + a(t).h(m^*,t)[X_q(t) - W_m(t-1)] \quad \text{si } m \in V(m^*,r(t)) \qquad (2.\ 49)$$

$$W_m(t) = W_m(t-1) \quad \text{si } m \notin V(m^*,r(t)) \text{ et } m \neq m^* \qquad (2.\ 50)$$

où :

- $W_m(t)$: est le vecteur poids du groupe de connexions des neurones de la couche d'entrée vers le m^{ème} neurone de la couche de sortie, à l'itération *t*.
- *r(t)* est le rayon de voisinage ou d'interaction qui dépend du rang *t* de l'itération considérée.
- *a(t)* est le coefficient d'apprentissage. Il peut être une fonction hyperbolique, une fonction exponentielle ou aussi une fonction linéaire du paramètre d'itération *t*.
- m* est le neurone gagnant défini par :

$$m^* = Arg \min_m [d(X_q(t), W_m(t))] \qquad (2.\ 51)$$

- $h(m^*,t)$ est la fonction d'interaction défini par :

$$h(m^*,t) = \exp\left(-\frac{d^2(U_m, U_{m'})}{2r^2(t)}\right) \qquad (2.\ 52)$$

Brown *et al.*[83] ont choisi de travailler avec un espace de couleur à deux dimensions (*r*, *g*). L'architecture de la carte pourrait donc être rectangulaire ou hexagonale, mais ils ont opté pour une architecture hexagonale. La figure 2.12 montre un exemple de deux cartes ainsi que les voisins de leurs nœuds centraux.

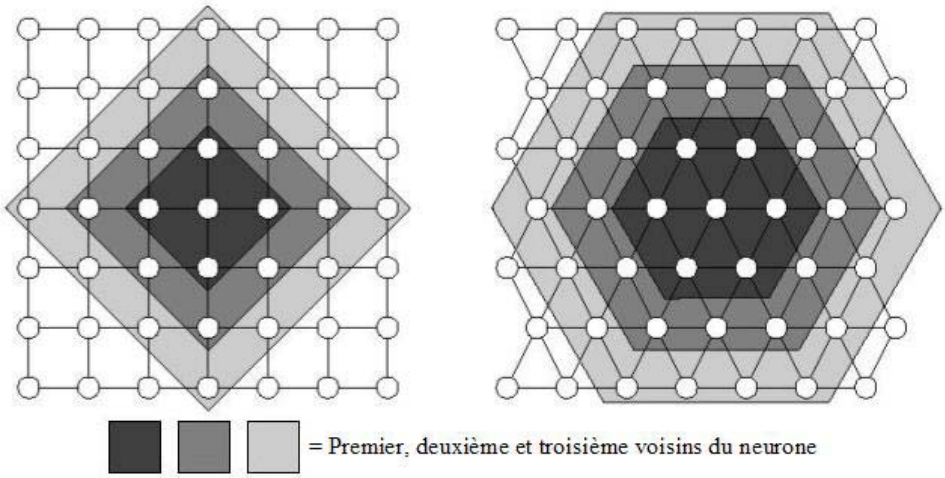


Figure 2. 12. *Architecture possibles de la carte SOM : rectangulaire (à gauche) et hexagonale (à droite)*

Ces auteurs ont effectué une série de tests afin de comparer leur modèle avec deux autres modèles : (1) modèle de mélange de gaussienne et (2) modèle basé sur les histogrammes décrit dans [80]. Ils ont prouvé que l'algorithme SOM est légèrement meilleur que le premier modèle, alors qu'il est inférieur au deuxième au niveau de la classification de pixels. L'avantage de leur méthode est qu'elle consomme moins de ressource que les 2 méthodes citées.

2.4.4 Autres modèles basée sur la couleur

Dans des applications spécifiques telles que le suivi de visage, ou encore le filtrage des images de nudités, la détection de peau est juste une étape préalable et devrait donc être rapide. Une méthode pour établir un classifieur de peau rapide est de définir explicitement (par un certain nombre de règles) les bornes des régions de peau dans un espace de couleur. A titre d'exemple Peer *et al.* [84] considèrent un pixel comme pixel de couleur de peau si chacune des conditions suivantes est respectée :

$$\begin{cases} R > 95, G > 40, B > 20 \\ \max\{R,G,B\} - \min\{R,G,B\} > 15 \\ |R - G| > 15,\ R > G, R > B \end{cases} \quad (2.\ 53)$$

La simplicité de cette méthode a attiré beaucoup de chercheurs [92], [93], [84], [85], [20], [21] et [96]. Son avantage est la simplicité des règles de décision utilisées pour discriminer les pixels de peau de ceux de non peau, ce qui produit une classification très rapide. Néanmoins, l'inconvénient majeur consiste dans le choix de l'espace de couleur ainsi que les règles et les

seuils de décision par des études empiriques. Notre objectif est de résoudre ces problèmes par un choix automatique le plus adéquat de l'espace de couleur ainsi que des règles de décision et des seuils.

2.5 Espaces de couleur utilisés pour la modélisation de la peau

Dans les systèmes de vision, le choix d'un système adapté de la représentation de la couleur est un problème très délicat [114]. L'espace de couleur le plus fréquemment utilisé est celui de RGB. Il décrit la couleur comme la corrélation de trois couleurs primaires (Rouge-Verte-Bleue). Chaque pixel d'une image couleur contient donc les trois composantes qui définissent sa couleur (cf. figure 2.13). C'est le principe de la vision trichromatique.

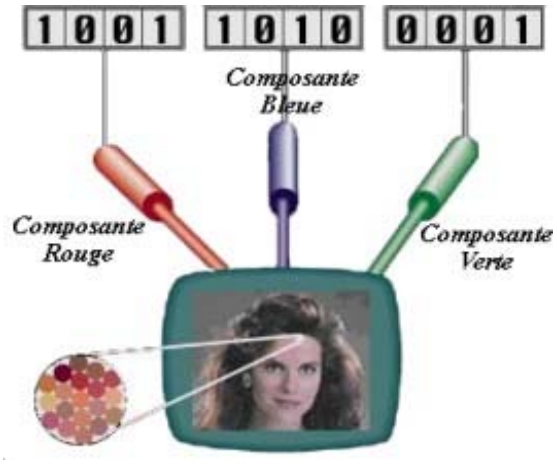


Figure 2. 13. *Représentation de la couleur*

D'autres espaces de représentation de couleurs peuvent être dérivées par des combinaisons linéaire ou non linéaire des composantes R, G et B. Ces espaces de couleur ont été proposés dans différentes applications [103] dont plusieurs ont été appliqués au problème de la modélisation de la couleur de la peau.

Dans les travaux [99] [100] [78] [80], l'espace de couleur RGB a été utilisé pour la segmentation d'image de couleur de peau. L'espace de couleur RGB normalisé est aussi populaire. De nombreux articles recommandent une détection de peau dans cet espace [19], [101] [23] [82] [83] [104] [66] [67].

L'espace de couleur HSV (Hue, Saturation Value), ainsi que ses variantes, HSI (Hue, Saturation, Intensity) et HLS (Hue, Lightness, Saturation) ont été utilisés fréquemment dans les travaux [71] [88] [89] [85] [21] [91] et [63]. De très nombreux articles [106] [107] [68] [81] et [20] affirment que l'espace de couleur YCbCr permet une bonne détection des pixels de la peau. Dans une moindre mesure, certains articles affirment qu'une bonne détection peut se faire dans d'autres espaces de couleur, tels que les espaces YIQ [78] [109], YUV [97] [108], XYZ de CIE [95], CIE L*a*b* par [102].

Nous signalons que la plupart des stratégies de sélection de l'espace de couleur visent à utiliser un espace 2D particulièrement utile quand les données d'apprentissage sont dispersées

ou quand le chercheur veut définir, par interactivité, la région de couleur de peau empiriquement [92] [93] [84] [85] [21] [96].

Des tests comparatifs, sur le meilleur espace de couleur permettant de détecter les zones de peau dans une image, ont été effectuées par Nicolas MOTTIN [16]. Cinq espaces de couleur différents ont été testés : YUV, RGB, HSI, TSL, et Lab. Les tests ont été faits sur la base de peau de Von Luschan (cf. figure 2.14), qui contient 36 imagettes de peau depuis les peaux les plus claires jusqu'aux peaux les plus mates. Ces résultats ont montré que la meilleure détection de peau peut se faire dans l'espace HSI.

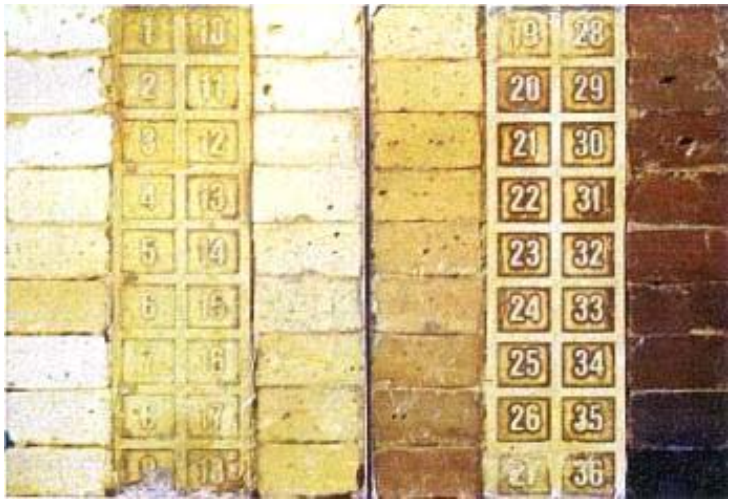


Figure 2. 14. *Base de peaux de Von Luschan*

Une autre étude faite par Girondel [24] montre que l'utilisation de l'espace 3D RGB (ou r, g, b) est troublante. Il l'a testé sur les imagettes de la base de peaux de Von Luschan dans l'espace *r-g*, et a obtenu une distribution assez étalée. Il a utilisé aussi 4 seuils (2 pour chaque composante normalisé) qui découlent de la distribution observée, mais il n'a obtenu que des résultats très médiocres. Il n'a pas été possible, selon lui, de trouver des seuils ou une modélisation pour la distribution obtenue donnant des résultats satisfaisants.

Dans l'article [67] apparaissent des seuils qui résultent de tests sur un millier de visages et sont censés donner de bons résultats pour la détection de peau dans l'espace RGB. Ces seuils ont été obtenus d'une modélisation gaussienne de chaque composante.

Seuils pour l'ensemble de la base de données :

$$m_R = 234.29 \quad \sigma_R = 26.77$$
$$m_G = 185.72 \quad \sigma_G = 30.41$$
$$m_B = 151.11 \quad \sigma_B = 25.68$$

Cet article donne aussi (pour l'ensemble des visages de la base de données) des seuils pour l'espace *rg*, qui montrent l'utilité de la normalisation :

$$m_r = 104.22 \quad \sigma_r = 4.93$$
$$m_g = 81.59 \quad \sigma_g = 3.89$$

Comme nous pouvons le constater, ces seuils varient sur une large gamme de valeurs. En adaptant ces deux seuils, les résultats obtenus par [67] sur l'ensemble de leurs séquences sont extrêmement surprenants : sur certaines séquences, aucune zone de peau n'est détectée alors que sur d'autres, de grandes zones contenant ou non de la peau sont détectées comme peau.

Nous signalons aussi que la performance d'un modèle paramétrique de peau dépend fortement du choix des espaces de couleur appropriés [71],[98]. D'autre part, les méthodes non paramétriques, utilisent souvent une base d'apprentissage importante et ne montrent pas une grande différence en choisissant les différents espaces de couleur ([82] [83] et [105]).

2.6 Performance des techniques existantes

Pour l'évaluation des performances des différents modèles de couleur de peau, des conditions de test doivent rester identiques. Malheureusement, il existe de nombreuses méthodes de détection de peau qui fournissent des résultats en se basant sur leurs propres bases de données qui ne sont malheureusement pas accessibles. La base de données d'images la plus célèbre pour l'apprentissage et les tests est celle de Compaq [70]. Le tableau 2.1 présente les meilleurs résultats de différentes méthodes sur cette base d'images : les taux de vrais positifs (VP) et les taux de faux positifs tels qu'ils sont cités par les auteurs.

Nous rappelons que :

- VP : c'est la probabilité qu'un pixel appartenant à la classe peau soit affecté à la classe peau
- FP : c'est la probabilité qu'un pixel appartenant à la classe non peau soit affecté à la classe peau

A partir de ces deux taux on peut déduire encore :

- FN : c'est la probabilité qu'un pixel appartenant à la classe peau soit affecté à la classe non peau
- VN : c'est la probabilité qu'un pixel appartenant à la classe non peau soit affecté à la classe non peau

Tableau 2. 1. *Performance de différentes méthodes de détection de peau*

Méthode	**VP** (Vrais Positifs)	**FP** (Faux Positifs)
Modèle bayésien basé sur les histogrammes (RGB) [80] [Jones and Regh] [80] [Jones and Regh] [78] [Brand and Mason]	 80% 90% 93.4%	 8.5% 14.2% 19.8%
Modèle basé sur une mixture de Gaussienne (RGB) [70] [Jones and Regh] [70] [Jones and Regh]	 80% 90%	 9.5% 15.5%
SOM en TS [83] [Brown and al.]	78%	32%

Modèle elliptique de borne (CIE-xy) [98] [Lee and Yoo]	90%	20.9%
Simple gaussienne (CbCr) [98] [Lee and Yoo]	90%	33.3%
Mixture de Gaussienne (IQ) [98] [Lee and Yoo]	90%	30%
Seuils fixés en YIQ [78] [Brand and Mason]	94.7%	30.2%

Le tableau 2.1 montre que le modèle bayésien basé sur les histogrammes en RGB proposé par Jones et Regh [80] donne la meilleure performance en termes de classification de pixels de peau/non-peau (taux des faux positifs, taux des vrais positifs). En deuxième lieu, on trouve le modèle basé sur une mixture de gaussiennes qui utilise l'espace de couleur RGB. Les autres méthodes viennent après. De ce fait, nous avons choisi de comparer nos résultats avec ceux obtenus par Jones et Regh [80] (Compaq).

2.7 Discussion et conclusion

L'état de l'art montre qu'il existe dans la littérature de nombreuses méthodes pour la détection de régions de peau dans une image. Trois approches se détachent principalement : la détection basée sur l'extraction de traits caractéristiques du visage et /ou des mains, la détection basée sur une approche couleur, et la détection basée sur le mouvement. Cette dernière n'est jamais utilisée seule.

Les techniques basées sur l'extraction de traits caractéristiques nécessitent la définition de modèles pour ces traits (main, visage, corps humain…) qui sont associés à des templates. Le template matching consiste à comparer l'intensité de pixels entre un template prédéfini et plusieurs sous régions de l'image que l'on désire analyser. Cette méthode s'effectue par plusieurs balayages sur l'ensemble de l'image. Les endroits les plus propices à la présence de l'objet cherché sont identifiés par une différence minimale entre le template et l'image. Cette approche souffre de deux handicaps : la variation de formes d'un objet et une connaissance supposée de l'échelle de l'objet. Pour pallier ces handicaps, divers templates peuvent être définis, mais la gestion des résultats obtenus peut s'avérer complexe. Ceci implique donc un balayage de l'image à diverses échelles, ainsi qu'un filtrage multirésolution d'un objet. De ce fait l'utilisation d'un template plus ou moins adapté au type d'objet recherché peut nuire à une détection efficace et diminuer la précision des résultats.

Dans le cas par exemple de détection de visage, différentes caractéristiques peuvent être déduites à partir des positions correspondantes sur le template. Celles-ci sont déterminées au préalable manuellement en positions relatives par rapport aux dimensions du template. Néanmoins, le template peut ne pas être parfaitement positionné en translation, en échelle et en rotation sur le visage à détecter. Dans ce cas, les coordonnées déduites sont légèrement erronées.

De plus, les techniques basées sur l'extraction de traits caractéristiques utilisent généralement des modèles déformables, des snakes ou des Point Distributed Models (PDM). Ces derniers requièrent une bonne résolution de l'image et sont difficilement réalisables en temps réel [46].

Les techniques basées sur le mouvement, telle que la méthode de différence d'image sont des techniques simples permettant de faire rapidement une estimation de la position d'un objet en mouvement. Cette estimation permet de réduire la zone de recherche d'un autre algorithme. On trouve cette approche souvent utilisée conjointement avec l'information de couleur de peau ou couplée avec un système basé sur la reconnaissance de traits caractéristiques. Cependant cette technique impose des contraintes sur l'environnement :

1. La caméra doit être fixe sous peine de détecter l'image entière comme objet en mouvement. Cette contrainte ne présente pas de difficulté dans des applications fixant une région particulière de la scène telle que le « bureau Numérique » de Wellner [25] [26] où seul le bureau doit être observé.
2. Les sources lumineuses doivent être constantes et fixes. Un changement de luminosité, même local, entraîne la détection de la zone de changement comme étant en mouvement. Un changement de luminosité peut être provoqué par l'allumage d'une lampe, le passage d'un individu créant une ombre ou le passage d'un nuage. Le problème est particulièrement crucial pour la différence avec une image de fond, puisque l'image a été prise dans des conditions particulières. Pour éliminer ce problème, une mise à jour de l'image de fond est nécessaire. La différence entre deux images successives étant moins sensible au changement de luminosité, l'hypothèse de conservation de l'intensité lumineuse devient valable [27].

Elle impose également des limitations :

1. En utilisant la technique de différence d'images successives et lorsque le mouvement est faible, la zone détectée est petite et ne contient qu'une partie de la zone cherchée.
2. Si deux objets sont en mouvement, un calcul supplémentaire est nécessaire pour différencier les deux objets. Ce calcul est simple lorsque les objets sont éloignés, un calcul de zones connexes est suffisant. Lorsque les objets sont superposés, il faut tenir compte d'autres critères tel que leur texture ou leur couleur.

Les objectifs que nous avons visés nous ont amené à choisir une méthode basée sur la détection de peau par une approche couleur. En effet, nous avons voulu que notre modèle de peau soit adapté pour les images fixes, ainsi que pour la vidéo. De ce fait, l'approche basée sur le mouvement ne correspond pas à notre besoin. A la différence de beaucoup de travaux qui ne traitent que des parties spécifiques d'un corps humain telles que les visages ou que les mains, nous visons ici un modèle général et robuste pour la détection de régions de peau dans une image par rapport à la diversité d'applications, de conditions de lumière et la diversité ethnique. Aussi, pour des raisons de temps de calcul, nous avons préféré l'utilisation de la même méthode dans le processus de détection. Ainsi, il nous est apparu qu'une méthode basée sur la détection de peau par une approche couleur est la plus adaptée à condition de bien choisir un espace de couleur approprié à la détection de peau.

Dans les méthodes basées sur la couleur, on distingue encore les méthodes paramétriques et les méthodes non paramétriques, ainsi que des méthodes basées sur une fixation empirique des règles et des seuils de décision. Pour les modèles paramétriques, leur pertinence et leur

qualité de précision dépendent, dans une large mesure, de la forme de distribution ainsi que de l'espace de couleur choisi. Par ailleurs, les phases d'apprentissage et de test pour ce type de méthodes sont lentes puisque celles-ci impliquent une procédure d'évaluation des paramètres tel que l'algorithme (EM) et l'évaluation des fonctions relativement complexes comme (2.29), (2.32) et (2.33).

Quant aux méthodes basées sur la fixation empirique des règles et des seuils de décision, elles sont des méthodes simples et rapides ce qui explique leurs utilisations pour des applications qui fonctionnent en temps réel. Néanmoins, on peut reprocher à ces méthodes la façon dont les règles de décision et les seuils sont choisis.

Les méthodes non paramétriques sont généralement rapides dans la phase d'apprentissage et de test. Elles ne font aucune hypothèse sur la répartition des données d'apprentissage et elles sont théoriquement indépendantes de la forme de distribution de la peau, contrairement aux méthodes paramétriques.

Nous avons donc opté pour une approche non paramétrique pour laquelle il existe encore deux alternatives : l'estimation non-paramétrique de la fonction de densité et l'estimation non-paramétrique de la fonction de classement. La première approche a été utilisée fréquemment dans la littérature pour la classification des pixels de peau et non peau. La seconde inclut les méthodes de classification par graphes d'induction. Cette dernière n'a pas été utilisée pour un tel type de classification.

Nous avons voulu faire un meilleur compromis entre le temps de traitement et la pertinence de l'analyse. C'est pourquoi nous avons opté pour l'utilisation des techniques de data mining en se basant sur les graphes d'induction. Ces derniers produisent des règles de décision efficaces et simples permettant ainsi une classification rapide et un traitement en temps réel.

Dans le chapitre suivant nous présentons le processus complet d'une classification supervisée par graphes d'induction qui est la base de nos travaux. Nous présentons aussi différentes approches de sélection de variables ainsi que les principaux algorithmes que nous avons utilisés.

Chapitre 3

Data Mining : Graphes d'induction

DATA-MINING : GRAPHES D'INDUCTION 41

3.1 Introduction

Les techniques de fouille de données ont été employées avec beaucoup de succès dans diverses applications telles que la gestion de relation client ou encore la gestion des connaissances [122]. Par ailleurs, face à l'explosion des technologies de l'information, de nouveaux types de documents se sont massivement répandus. Par exemple, le Web est un vecteur de forte diffusion de documents multimédias comme le texte, l'image et la vidéo [123]. Cet ensemble de données fortement hétérogènes qu'on peut appeler aussi données complexes nécessite une modélisation spécifique et des méthodes d'accès très avancées dans le processus de l'extraction de connaissances.
Cette thèse s'inscrit dans le cadre général du traitement de ces données complexes. Nous avons particulièrement utilisé les données issues des images pour la définition de notre modèle de peau, ainsi que des données textuelles combinées à celle des images pour la mise en application de notre logiciel de filtrage.

Nous présentons dans ce chapitre ce qu'est un problème de classification supervisée. En effet, dans certains cas, il est possible de décrire complètement, de manière linguistique, la démarche de classification ; dans ce cas, un algorithme reproduisant cette démarche peut être construit, et le problème est résolu. Dans d'autre cas, il est impossible de décrire précisément la classification ; une solution consiste alors à demander à un professeur (ou superviseur, expert) de classer un échantillon d'individus. Des méthodes de résolution qui « apprennent par l'exemple » sont capables de produire des règles de décision permettant de classer de nouveaux exemples inconnus. C'est à ces dernières méthodes que nous nous intéressons dans cette thèse.

3.2 Apprentissage supervisé

L'apprentissage automatique devient une préoccupation majeure de l'intelligence artificielle dès la fin des années 70, lorsque la vogue des systèmes experts se heurte à la difficulté d'acquérir l'expertise existante, ou de la constituer lorsqu'elle est inexistante. Depuis cette date la communauté scientifique a proposé une série de techniques et d'outils pour l'apprentissage. Nous les regroupons selon deux axes : la reconnaissance des formes et la fouille de données ou pour être plus précis, l'extraction de connaissance à partir de données. Le second domaine, né des années 1990 sous le nom de fouille de donnée ou data mining, est le moins connu des deux bien qu'il soit porteur de réels apports.

L'apprentissage vise à construire des hypothèses à partir d'exemples. Simon [128] l'interprète comme les changements dans un système qui accomplira au mieux la même tâche, ou une tâche similaire dans la même population dans l'avenir. Dietterich [129] propose une approche plus fonctionnelle qui permet d'évaluer un système de connaissance en le reliant à la notion de « connaissances ». Il distingue ainsi trois niveaux de description d'un système d'apprentissage : ceux ne recevant aucune entrée et accomplissant au mieux une tâche ; ceux qui reçoivent des connaissances via des entrées mais n'accomplissent aucune induction ; et enfin, ceux qui reçoivent des entrées et en extraient des connaissances qui ne sont connues ni implicitement ni explicitement, c'est l'apprentissage inductif.

Dans cette thèse nous nous intéressons à l'apprentissage inductif, plus particulièrement l'apprentissage empirique qui vise à produire des règles générales à partir d'une série d'observations [130]. Formellement nous caractériserons de la manière suivante l'inférence inductive : Soit D un domaine, composé d'une population Ω. Nous disposons d'un échantillon Ω^a et d'un algorithme d'apprentissage A. Sachant Ω^a et D, A produit une théorie M, issue de l'espace des hypothèses, que l'on peut utiliser pour expliquer la structure des données. Les objectifs peuvent être multiples : donner une description plus compacte des observations, distinguer les « structures » sous-jacentes qui régissent leur formation, prédire l'appartenance ou la valeur prise par un individu quelconque de la population originelle. L'inférence inductive recouvre deux domaines d'études qui ne sont pas nécessairement distincts selon le type d'information inféré : l'apprentissage non supervisé et l'apprentissage supervisé [126].

Dans l'apprentissage non supervisé, l'algorithme A utilise un vecteur d'attributs $\vec{X} = (X_1(.),...,X_p(.))$ pour essayer de trouver des « régularités » dans l'échantillon d'apprentissage. Elles se manifestent principalement par la constitution de groupes dans lesquels les observations diffèrent très peu au regard des valeurs prises par les $X_i(.)$. Ces variables peuvent être continues ou qualitatives (prenant leurs valeurs dans X_i (Ω) = $\{x_{i1},\ldots,x_{i2}\}$).

L'apprentissage supervisé vise toujours à partir d'un vecteur d'attributs $\vec{X}$ que l'on nomme ici *attributs prédictifs*, ou encore *variables exogènes* de reconstruire une fonction ou concept sous-jacent f telle que :

$$Y = f(\vec{X}) \qquad (3.\ 1)$$

Y (.) est qualifiée de *variable à prédire*, ou encore de *variable endogène*. L'apprentissage permet de mettre à jour un modèle M que l'on nomme *classifieur* ou *prédicteur*, tel que :

$$\hat{Y} = M(\vec{X}) \qquad (3.\ 2)$$

avec pour objectif $\hat{Y}(.) = Y(.)$

Selon la nature de Y(.), nous distinguons généralement deux familles d'apprentissage supervisé : lorsque Y(.) est continu, on parle de régression. Lorsqu'il prend ses valeurs dans un ensemble fini $\{y_1,\ldots,y_K\}$, l'espace des étiquettes ou encore les classes, on parle plutôt de classement, c'est le thème principale qui nous intéresse.

L'une des finalités de l'apprentissage supervisé est le diagnostic et la prévision. Prenons l'exemple d'identification des pixels de peau dans une image. La variable à prédire ici est la variable « classe du pixel », qui ne prend que deux modalités « peau » ou non peau.

Pour des raisons diverses telles les conditions d'éclairage, la diversité ethnique etc., l'identification du pixel de peau est un problème complexe. C'est la raison pour laquelle nous cherchons un moyen φ pour prédire la classe de pixel (peau /non peau).

Ce processus d'induction peut s'insérer dans une démarche plus générale d'extraction de connaissances à partir de données.

3.3 Extraction de connaissances à partir de données (ECD)

L'Extraction de Connaissances à partir de Données (ECD), communément appelée Data Mining, est un domaine aujourd'hui très en vogue. Nous allons d'abord donner quelques définitions générales de l'ECD. Nous décrivons ensuite les étapes principales d'une telle démarche.

On s'intéresse ici à la découverte de connaissances. On cherche donc des techniques d'exploration des données pour trouver des formes "intéressantes" qui aident à expliciter une information auparavant cachée dans les données. Les problèmes fondamentaux de la découverte de connaissances sont donc : la représentation des connaissances, la sélection des attributs, la prise en compte des données manquantes, bruitées ou rares, la découverte de formes "intéressantes", "utiles" ou encore "surprenantes" [132]. Comme Brachman et Anand [141] le soulignent, ce processus n'est pas une exploitation pure et simple, mais un processus compliqué plus proche de l'archéologie que de l'exploitation. Fayyad et al. la définit comme "un processus non-trivial d'identification de structures inconnues, valides et potentiellement exploitables dans les bases de données [125]". Cette définition est une des premières qui traite explicitement de l'ECD, par la suite plusieurs tentatives de re-définition sont apparues pour mieux préciser le domaine mais aucune ne s'est réellement imposée.

L'Extraction de Connaissances à partir de Données (ECD) est un processus complexe qui se déroule suivant une série d'opérations [125]. Nous pouvons regrouper ces opérations en trois étapes majeures. Elles sont la préparation des données, la fouille de données à proprement parler qui est l'étape centrale de l'ECD et enfin la validation des modèles ainsi élaborés :

1. L'étape de pré-traitements et de sélection des données : il s'agit dans cette phase de déterminer la structure générale des données ainsi que les règles utilisées pour les constituer. Il faut identifier les informations exploitables et vérifier leur qualité et leur efficacité d'accès afin de construire des tables bidimensionnelles, des corpus de données spécifiques. La recherche d'une sélection optimale d'attributs est le point central d'un processus de data mining, c'est elle qui va conditionner la qualité des modèles établis. En effet, la volonté d'intégrer toutes les variables à un niveau très fin entraîne un surdimensionnement du problème, qui nuit la capacité de généralisation. Cette capacité de généralisation permet à un modèle de conserver des performances comparables sur la base d'apprentissage et sur la base de test.

2. La fouille de données : c'est l'étape de recherche du modèle, qu'on appellera aussi phase de modélisation, qui consiste à extraire la connaissance utile d'un ensemble de données et à la présenter sous une forme synthétique. Il s'agit de la phase la plus souvent décrite sous le terme data mining et qui repose pour partie, sur une recherche exploratoire, c'est à dire dépourvue de préjugés concernant les relations entre les données. La fouille de données s'effectue généralement sur des tables bidimensionnelles et se décompose essentiellement en trois grandes familles de méthodes [124] :

a. Les méthodes descriptives qui sont principalement issues de la statistique descriptive et de l'analyse de données ;

b. Les méthodes de structuration qui regroupent toutes les techniques d'apprentissage non supervisé et de classification automatique ;

c. Les méthodes explicatives qui cherchent à établir un modèle décrivant un phénomène, défini à partir d'une variable endogène, à l'aide d'un ensemble de descripteurs appelés variables exogènes.

3. A l'issu de la construction du modèle, il est théoriquement possible d'en tester la pertinence de ce dernier sur la base d'apprentissage. Il est toutefois fréquent qu'on apprenne les données plutôt que le modèle. Il est donc préférable de constituer au préalable une base de test ne servant qu'au test. Cette dernière permet de valider le modèle et atteindre enfin le stade de connaissance.

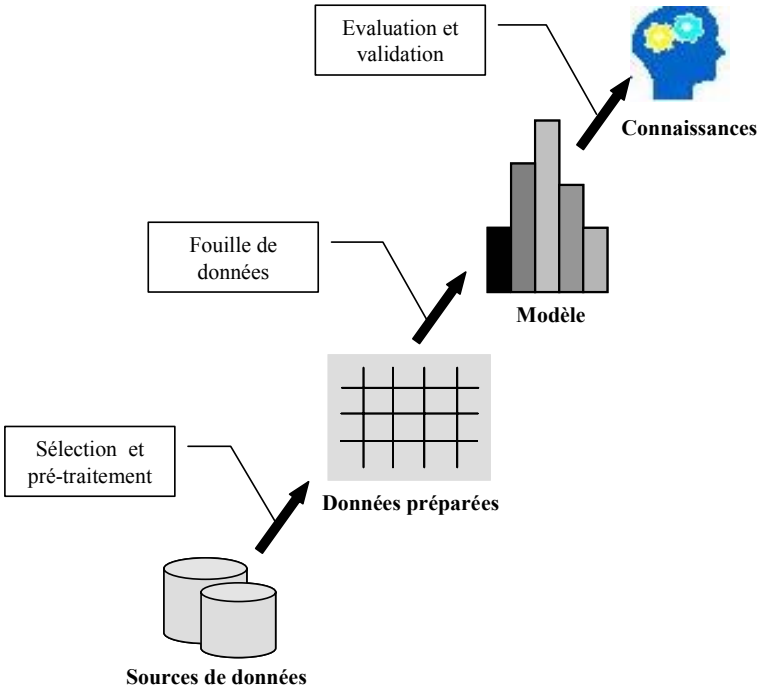


Figure 3. 1. *Processus d'extraction de connaissances à partir de données*

3.4 Qualités désirées d'un classifieur

Avant de présenter les techniques basées sur les graphes d'induction et afin d'argumenter notre préférence pour ces techniques, nous présentons dans cette section les qualités désirées d'un classifieur. La pertinence d'un classifieur peut être apprécier de différentes façons. On s'accorde pourtant à reconnaître l'importance de quelques critères globaux, qui sont les plus souvent cités dans les publications, ils ne doivent pas nous en masquer d'autres qui sont tout aussi importants selon le domaine sur lequel on travaille [142] [131].

3.4.1 La précision

C'est le critère certainement le plus important de l'apprentissage supervisé. Elle montre la capacité intrinsèque du classifieur à reconnaître la variable à prédire dans la population. Lorsque la précision est parfaite, c'est à dire zéro erreur, on peut penser à juste titre que l'on a trouvé une expression du concept à apprendre. Comme nous avons déjà dis la précision parfaite sur un fichier d'apprentissage ne reflète pas nécessairement une bonne qualité de prédiction. En effet c'est la précision sur toute la population qui nous préoccupe.

3.4.2 La compréhensibilité

Dans la définition de l'ECD l'exploitabilité de la connaissance en était un des principaux objectifs. Celle-ci passe par la compréhensibilité du modèle. Selon Michalski [143], la connaissance extraite doit être sémantiquement et structurellement similaire à celles qu'un expert humain peut produire. Cette compréhensibilité est avantageuse pour plusieurs motifs :

- L'appropriation : une connaissance est mieux mise en œuvre si elle est comprise et facile à accepter par ses utilisateurs.

- Le déchiffrage : l'exploration de données met souvent à jour des régularités que l'on ne soupçonne pas dans le domaine d'étude, soit parce que la formation de l'expert ne couvre pas cette partie de la connaissance, soit parce que temporellement de nouveaux phénomènes ont vu le jour. Quoi qu'il en soit, des modèles intelligibles permettent au chercheur de mieux déchiffrer les connaissances extraites pour pouvoir en juger la pertinence ;

- L'explication : prédire la classe d'un individu est une chose, expliquer pourquoi en est une autre. Un des reproches souvent adressés aux classifieurs « boîtes noires » est qu'ils fournissent une prédiction sans que l'on sache comment ils y sont parvenus. Notons que de gros efforts sont faits pour réduire l'opacité de certains modèles notamment dans les modèles connexionnistes à l'instar de réseaux de neuronne qui souffrent de ce handicap [144] [145] ;

- La rétroaction : la tâche d'apprentissage n'est jamais définitive, de nouvelles informations ainsi que des connaissances du domaine peuvent nous amener à améliorer manuellement un classifieur afin d'en augmenter les performances. Cette manipulation n'est possible que si nous puissions appréhender les éléments qui la composent, et apprécier qualitativement les modifications introduites ;

- La flexibilité : aucun algorithme ne détient la vérité absolue. Sur un domaine donné, certains algorithmes peuvent marcher mieux que d'autres, plusieurs algorithmes peuvent également mettre à jour différentes facettes d'une connaissance. Dans cette optique, l'agrégation des classifieurs d'origines différentes semble être une voie intéressante [146]. Il est certainement plus facile de synthétiser plusieurs prédicteurs s'ils sont individuellement compréhensibles.

Ces différentes raisons éclairent chacune un aspect particulier de la compréhensibilité du modèle, mais finalement se rejoignent, surtout en ce qui concerne les graphes d'induction, dans la notion de complexité. Plus un modèle est complexe moins on aura de facilité à le comprendre et inversement. Dans un cadre plus général, il est évident que l'on saisit mieux la partition de l'espace de représentation d'un perceptron simple que le découpage effectué par un perceptron multicouche, cette opacité augmentant avec le nombre de couches cachées[131].

3.5 Les graphes d'induction

Les graphes d'induction, plus connus sous le terme d'arbres de décision, tiennent une place particulière car ils réalisent selon Clark [147] le meilleur compromis entre performances en précision et compréhensibilité.

Un arbre de décision est la représentation graphique d'une procédure de classification. Il s'agit vraiment d'un arbre au sens informatique du terme. En effet, à toute description complète est associée une seule feuille de l'arbre de décision. Cette association est définie en commençant par la racine de l'arbre et en descendant dans l'arbre selon les réponses aux tests qui étiquettent les nœuds internes. La classe associée est alors la classe par défaut associée à la feuille qui correspond à la description

3.5.1 Définitions générales et notations

Soit Ω l'ensemble d'individus concernés par un problème d'apprentissage. A cette population est associé un attribut particulier appelé "attribut classe" qu'on noté *C*. Il s'agit en fait d'une variable statistique appelée dans le domaine de statistique "*variable endogène*" ou simplement "classe".

À chaque individu *w* est associé sa classe *C*(*w*). On dit que la variable *C* prend ses valeurs dans l'ensemble des étiquettes, appelé également "ensemble de classe" et noté ¢.

$$C : \Omega \to ¢ = \{c_1, c_2, ...c_m\}$$
$$w \to C(w)$$

Dans la réalité, l'observation de *C*(*w*) n'est pas toujours facile et ceci pour des raisons diverses. C'est la raison pour laquelle nous cherchons un moyen φ pour prédire la classe *C*.

La détermination du modèle de prédiction φ est liée à l'hypothèse selon laquelle les valeurs prises par la variable statistique *C* ne relèvent pas du hasard, et q'elles suivent certains critères que l'on peut caractériser. Pour cela, l'expert du domaine concerné établit une liste a priori de variables statistiques, appelée "*variables exogènes*" et notées : X= (X_1, X_2, ..., X_p). Ces variables sont également désignées par des terminologies différentes, par exemple "*attributs prédictifs*", ou encore "*attributs explicatifs*"

Les variables exogènes prennent leurs valeurs dans un espace de représentation notée ℜ qui ne possèdent pas de structure mathématique particulière a priori :

$$X : \Omega \to \Re$$
$$\omega \to X(w) = \left(X_1(w), X_2(w), ..., X_p(w)\right)$$

L'objectif est de rechercher un modèle de prédiction φ permettant, pour un individu w issu de Ω dont nous ne connaissons pas la classe $C(w)$ mais connaissons l'état de toutes les variables exogènes $X(w)$, de prédire cette valeur grâce à φ. Nous souhaitons que φ possède la propriété de cohérence définie comme suit.

Un prédicteur φ est dit *cohérent* si pour deux individus w et w' nous observons la même valeur $\varphi(w) = \varphi(w')$, alors ils doivent nécessairement appartenir à la même classe ; $C(w) = C(w')$.

En général, cette propriété de cohérence ne pourra être vérifiée qu'a posteriori. Dans la pratique, nous construisons φ sur un échantillon d'apprentissage $\Omega_a \subset \Omega$ et nous contentons de la vérifier sur une majorité d'individus issus d'un échantillon de test $\Omega_t \subset \Omega$, différent de l'échantillon d'apprentissage Ω_a. Ainsi, pour tout individu $w \in (\Omega_t \cup \Omega_a)$, nous proposons connues à la fois ses valeur $X(w)$ dans l'espace de représentation $\Re$ et sa classe $C(w)$ dans l'espace des étiquettes $\not\in$.

Si φ est jugée cohérente, alors nous pouvons généraliser son emploi à tous les individus de la population Ω. Ainsi grâce à φ, nous pourrons calculer $C(w)$, pour tout individu $w \in \Omega - (\Omega_a \cup \Omega_t)$, connaissant $X(w)$.

3.5.2 Principe de construction des graphes

Les graphes d'induction exploitent largement le concept de partition d'un ensemble. On appelle partition engendrée par C sur Ω, la partition :

$$P_{C(\Omega)} = \left\{\Omega_{c_1}, \Omega_{c_2}, \ldots, \Omega_{C_m}\right\} \quad (3.3)$$

définie par :

$$\Omega_{c_k} = \left\{w \in \Omega; C(w) = c_k\right\} \quad (3.4)$$

Ω_{c_k} est l'ensemble des individus de la population Ω appartenant à la classe C_k.

En apprentissage, notre objectif est de construire un modèle φ nous permettant d'obtenir, au moyen des variables $X_1, X_2, ..., X_p$, une partition de Ω. Nous serons amenés à comparer la partition engendrée par C à celle engendrée par φ. Si les deux partitions sont identiques nous conclurons alors que φ est un bon modèle pour déterminer C.

Le principe de construction d'un graphe d'induction est de diviser récursivement et le plus efficacement possible les exemples de l'ensemble d'apprentissage par des tests définis à l'aide des attributs prédicatifs jusqu'à ce que l'on obtienne des sous-ensembles d'exemples ne contenant (presque) que des exemples appartenant tous à une même classe. On part donc de la partition grossière située à la racine de l'arbre. On cherche parmi les p variables, celle qui

donne la "meilleure nouvelle partition" au sens d'un critère qui diffère selon l'algorithme utilisé. Sur chaque élément de cette partition, on répète le processus de segmentation comme s'il s'agissait de la racine, sans se préoccuper de ce qui se passe sur les autres sommets de l'arbre.

Dans toutes les méthodes, on trouve les trois opérateurs suivants :

1. **Décider si un nœud est terminal**, c'est-à-dire décider si un nœud doit être étiqueté comme une feuille.
2. **Sélectionner un test à associer à un nœud**. Par exemple : aléatoirement, utiliser des critères statistiques, ...
3. **Affecter une classe à une feuille**. On attribue l'étiquette de la classe majoritaire sauf si l'on utilise des fonctions coût ou risque.

Les méthodes différent par des choix effectués pour ces différents opérateurs, notamment sur le choix d'un test (par exemple, utilisation du gain et de la fonction entropie) et le critère d'arrêt (quand arrêter la croissance de l'arbre, soit quand décider si un nœud est terminal). Généralement, on déclare qu'un nœud est terminal s'il n'existe aucune variable qui permet d'engendrer localement une sous-partition qui puisse améliorer la valeur du critère utilisé. Le processus s'arrête si tous les nœuds sont saturés. La condition de saturation peut être enrichie par d'autres paramètres comme l'introduction d'une contrainte d'admissibilité sur les nœuds, qui permet d'éviter l'apparition de nœuds dont les effectifs seraient trop faibles pour être significatifs sur le plan statistique.

Le schéma général des algorithmes est le suivant :

Algorithme 3.1: Algorithme générique de construction d'un arbre de décision

```
Données : Données d'apprentissage
Résultat : Arbre de décision
Initialiser : arbre vide ;
Nœud courant: racine ;
répéter
  décider si le nœud courant est terminal
      si le nœud est terminal alors
          Affecter une classe à ce nœud ;
      sinon
          Sélectionner un test et créer le sous-arbre ;
      finsi
  nœud courant : nœud suivant non exploré s'il en existe ;
Jusqu'à obtention de l'arbre de décision
```

Nous pouvons représenter les partitions de Ω engendrées par les variables $X_1, X_2, ..., X_p$ au moyen d'un graphe latticiel $\Gamma = (\Sigma, A)$ appelé aussi graphe d'induction.

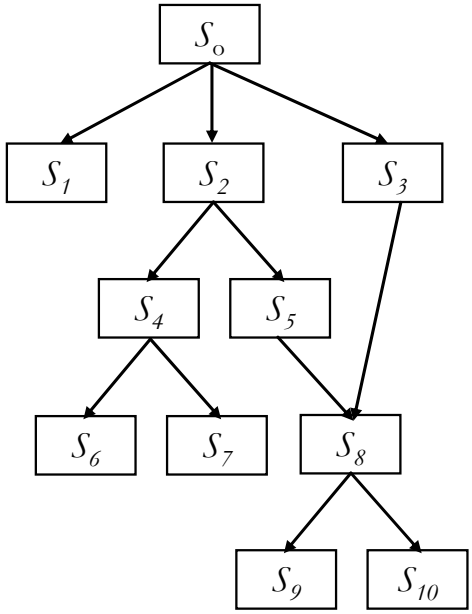


Figure 3. 2. *Structure d'un graphe d'induction*

Dans un graphe d'induction, les nœuds internes sont appelés *nœuds de décision*. Un tel nœud est étiqueté par un *test* qui peut être appliqué à toute description d'un individu de la population. En général, chaque test examine la valeur d'un unique attribut de l'espace des descriptions. Les réponses possibles au test correspondent aux labels des arcs issus de ce nœud. Chaque chemin correspond donc à une règle exprimée généralement dans le formalise de la logique des propositions sous la forme "si *condition* alors *conclusion*" dans laquelle "*condition*" désigne une disjonction de conjonctions de propositions logiques de type "attribut, valeur".

3.6 Sélection de variables en classification

Cette section traite une question très importante en classification, et plus généralement dans tout problème de modélisation statistique : celle du choix des variables pertinentes parmi un ensemble de variables. Ce choix constitue une étape capitale dans la construction du modèle de prédiction φ, et a des conséquences majeures sur ce dernier. En effet, la sélection des variables non ou faiblement pertinentes peut réduire la compréhension et les performances de reconnaissance d'un modèle. Une étude présentée dans les travaux de Trun et al. [133] montre que la non suppression d'une variable non pertinente entraîne la génération d'arbres de décision plus profonds et moins performants avec l'algorithme C4.5 que ceux obtenus sans cette variable. Aha et al. [134] montrent d'ailleurs que le stockage de l'algorithme ID3 augmente exponentiellement avec le nombre de variables non pertinentes. Donc, d'un point de vue pratique, la sélection d'un bon sous-ensemble de variables est capitale sur les performances du classifieur et la complexité du modèle.

3.6.1 Définition de la sélection de variables

La sélection de variables consiste, dans un problème d'extraction de connaissances où une variable représente un élément descriptif d'un objet, à réduire l'ensemble des variables

considérés. Ceci peut augmenter la précision de la prédiction ou réduire le temps de traitement des données. Classiquement, la sélection de variables est définie comme le fait de sélectionner un sous-ensemble de M attributs à partir d'un ensemble N, tel que $M < N$ et que la fonction critère choisie soit optimale sur le sous-ensemble de taille M choisi. Une procédure de recherche est donc mise en place dans le but d'explorer l'espace de tous les sous-ensembles de variables. La performance de chaque sous-ensemble est calculée grâce à une fonction d'évaluation. La sortie est le sous-ensemble qui optimise cette fonction d'évaluation.

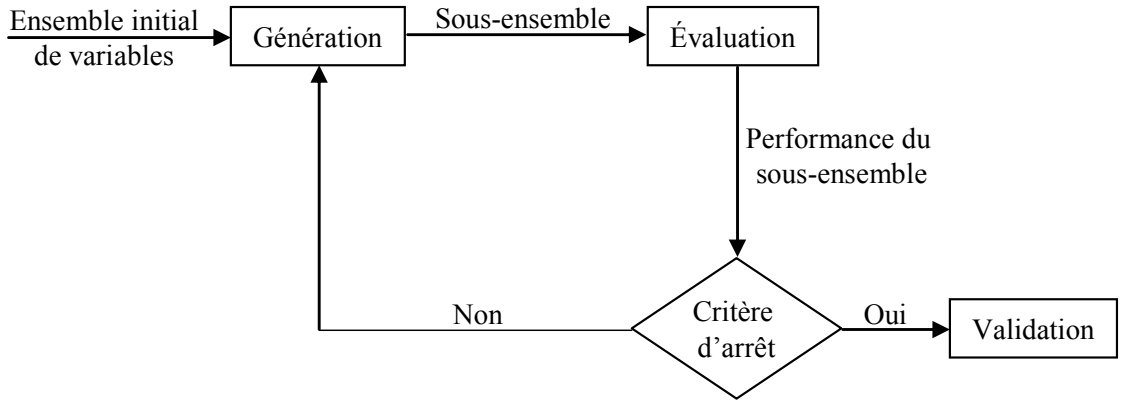


Figure 3. 3. *Processus de sélection de variables*

Après avoir sélectionné un sous-ensemble de variables, l'ensemble initial des données d'apprentissage est réduit. Ensuite l'algorithme d'apprentissage est mis en œuvre. Enfin, les performances de la classification sont évaluées sur les résultats de l'algorithme d'apprentissage à partir de données tests.

La sélection d'attributs essaye de sélectionner le sous-ensemble le plus petit selon le critère suivant : l'exactitude de la classification ne doit pas diminuer de manière significative. La sélection de variables peut également être vue comme un compromis entre le nombre de variables et la qualité prédictive : à nombre de variables égal, il faut prédire le mieux possible, et à taux de prédiction égal, il faut avoir le plus petit nombre de variables possible.

3.6.2 Méthodes de sélection de variables

Il existe deux approches visant à sélectionner un sous-ensemble minimum de variables : l'approche de type enveloppe (Wrapper Approach) et l'approche de type filtre (Filter Approach). La différence fondamentale entre ces deux approches réside dans le fait que la première est liée à l'algorithme d'induction utilisé alors que la seconde est totalement indépendante. La figure 3.4 montre le processus de sélection selon l'approche filtre, et la figure 3.5 celui selon l'approche enveloppe.

Les méthodes filtrantes sélectionnent des variables en utilisant différentes approches et différents critères pour calculer la pertinence d'une variable avant le processus d'apprentissage, c'est à dire avant la construction du classifieur.

Les méthodes enveloppantes se servent de l'algorithme d'induction : l'apprentissage est effectué avec les variables sélectionnées et les performances sont estimées à partir de l'erreur de généralisation

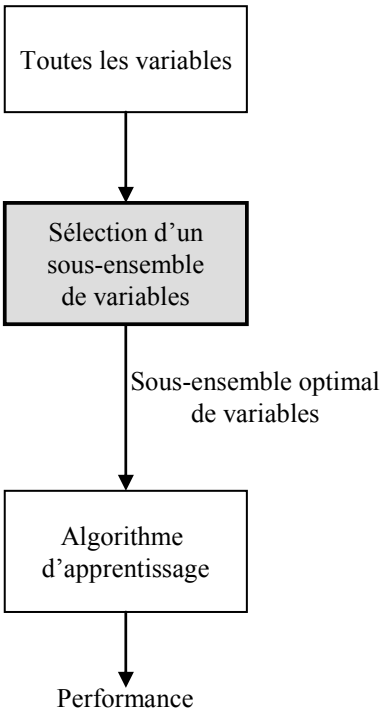


Figure 3. 4. *Approche Filtre*

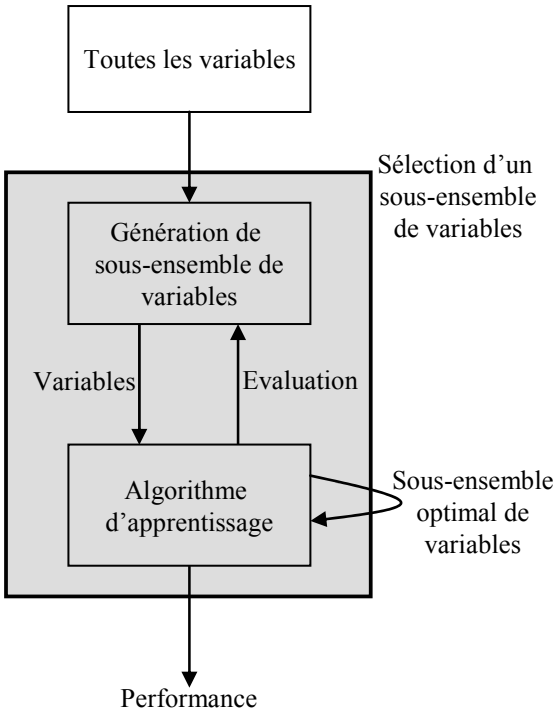


Figure 3. 5. *Approche Enveloppe*

La méthode enveloppante conduit une recherche dans l'espace des paramètres possibles. Une recherche requiert [140] :

- Un espace d'états où chaque état représente un sous ensemble d'attributs. Pour *n* attributs, il y a *n* bits dans chaque état et chaque bit indique si l'attribut est présent ou absent.
- Un état initial : lorsque l'on fait une sélection "forward", la recherche commence avec un ensemble vide d'attributs ; lorsque l'on fait une élimination "backward", la recherche commence avec l'ensemble complet d'attributs.
- Une condition d'arrêt.
- Une méthode de recherche.

Les méthodes enveloppantes les plus utilisées sont :

- *Méthode SFS (Sequential Forward Selection)* : la méthode SFS est de conception ascendante (bottom-up). Partant d'un ensemble vide on sélectionne la meilleure caractéristique, celui qui maximise le gain d'incertitude. Puis à chaque itération, on choisit une caractèristique qui jumelée à celles préalablement trouvées permet une discrimination maximale.

- *Méthode SBS (Sequential Backward Selection)* : SBS commence avec tous les attributs et enlève un attribut à chaque fois (celui qui, étant enlevé, conduit à une amélioration maximale).

3.7 Algorithmes de génération de graphe d'induction

Nous étudions dans cette section les algorithmes de génération d'arbres de décision à partir de données. Les algorithmes les plus répandus et les plus utilisés dans le domaine de fouille de données sont CART (Classification And Regression Trees), ID3, C4.5, et SIPINA, La notoriété de ces méthodes est telle qu'il paraît difficile de faire un article sur les arbres et graphes d'induction sans citer au moins une de ces méthodes.

3.7.1 Algorithme CART

CART (Classification And Regression Trees)[150], est une méthode très populaire. Il constitue un cadre méthodologique pour générer des graphes d'induction binaires.

Soit un échantillon d'apprentissage Ω_a . Tout individu $\omega \in \Omega_a$ est décrit par p variables statistiques notées $X_1, X_2, ..., X_j, ..., X_p$ et appartient à une classe $C(\omega)$ parmi m notées $c_1, c_2, ..., c_k, ...c_m$. On suppose que le nombre d'individus qui composent Ω est égal à n. On notera $X_j(\Omega) = \{x_{j1}, x_{j2}, ..., x_{jk}, ..., x_{j\alpha_j}\}$ l'ensemble des valeurs distinctes prises par X_j.

Il s'agit de construire une succession de partitions, que nous pouvons schématiser par un arbre binaire.

La racine S_0 contient la partition grossière Ω_a. Ensuite, à partir d'une variable X_j, choisie parmi p variables, on construit une bi-partition sur Ω_a notée $S_1 = \{s_{g_1}, s_{d_1}\}$ où $s_{g_1} = \{\omega \in \Omega_a ; X_j(\omega) \in X_j\}$ correspond à la sous population de la branche gauche et $s_{d_1} = \{\omega \in \Omega; X_j(\omega) \notin X_j\}$ à celle de la branche droite. Il faudra néanmoins optimiser cette bi-partition. Les éléments s_{g_1} et s_{d_1} sont ensuite segmentés en deux avec une autre variable. Ce processus est renouvelé sur chacun des sommets terminaux jusqu'à son arrêt selon certaines conditions.

Le développement de l'arbre s'effectue de façon indépendante à partir de chaque sommet terminal. Soit s le sommet considéré à partir duquel nous générons la bipartition $S = \{s_g, s_d\}$. Supposons que ces deux sommets ont respectivement $n_g = card(s_g)$ et $n_d = card(s_d)$ individus. Soient n_{ig} et n_{id} les effectifs de la classe c_i, respectivement dans s_g et s_d. Deux critères sont utilisés pour choisir la meilleure segmentation :

- Le critère de Gini surtout utilisé pour les problèmes à deux classes ($m = 2$) : ici on cherche la bi-partition $\{s_g, s_d\}$ qui minimiser l'indice d'impureté de Gini donné par la formule suivante :

$$I(s_g, s_d) = \frac{n_g}{n} \sum_{i=1}^{m} \frac{n_{ig}}{n_g} \left(1 - \frac{n_{ig}}{n_g}\right) + \frac{n_d}{n} \sum_{i=1}^{m} \frac{n_{id}}{n_d} \left(1 - \frac{n_{id}}{n_d}\right) \qquad (3.5)$$

Il est équivalent à la bi-partition qui maximise la variation d'impureté ou le gain informationnel $\Im_G(s_g, s_d)$:

$$\Im_G(s_g, s_d) = \sum_{i=1}^{m} \frac{n_{ig} + n_{id}}{n} \left(1 - \frac{n_{ig} + n_{id}}{n}\right) - I(s_g, s_d) \qquad (3.\ 6)$$

ou

$$\Im_G(s_g, s_d) = \frac{n_g}{n} \frac{n_d}{n} \sum_{i=1}^{m} \left(\frac{n_{ig}}{n_g} - \frac{n_{id}}{n_d}\right)^2$$

- Le critère Twoing s'applique davantage aux problèmes à plusieurs classes ($m > 2$) et il est donné par la formule suivante :

$$\Im_T(s_g, s_d) = \frac{\frac{n_g}{n} \frac{n_d}{n}}{4} \left[\sum_{i=1}^{m} \left|\frac{n_{ig}}{n_g} - \frac{n_{id}}{n_d}\right|\right]^2 \qquad (3.\ 7)$$

L'algorithme consiste à la construction d'un arbre binaire comme suit :

Algorithme 3.2 : Algorithme CART

1. Choix du critère pour mesurer la qualité de la partition.
2. i : indice pour parcourir les sommets, i=0 on est à la racine de l'arbre
3. Détermination de la variable donnant la meilleure bipartition au sommet s_t .
4. Si pour la meilleure bipartition, on a un gain $J_T(s_{gt}, s_{dt}) > \beta$ (où β est fixé généralement à zéro), on continue la bipartition avec les deux nouveaux sommets. On peut aussi ajouter la contrainte τ correspondant à l'effectif minimal d'une partition.
5. Le processus s'arrête lorsqu'il n'y a plus d'amélioration.
6. Chaque sommet est affecté à une classe (spécialisation).

3.7.2 Algorithme ID3

L'algorithme Induction Decision Tree, plus communément appelé ID3 [148], est l'une des plus anciennes méthodes de construction de règles d'induction. La construction du graphe se fait en recherchant les attributs qui apportent le plus d'information au regard d'un critère propre à l'algorithme. Ce critère peut être éventuellement modifié et il est possible d'ajouter certaines conditions d'arrêt pour le processus de partitionnement d'un sommet. Ce type de modifications a donné naissance à C4.5 [149]. Cet algorithme a lui-même donné naissance à Improved C4.5

Pour ID3, le processus de construction du graphe est le suivant : on part d'une partition grossière et on cherche parmi les variables celle qui donne la meilleure partition. On répète le processus de segmentation pour chaque élément de la nouvelle partition. Dans ID3 le passage d'une partition S_i à une autre S_{i+1} se fait seulement par segmentation de l'un de sommets terminaux de l'arbre en maximisant le gain d'incertitude $\Im(S_{i+1})$, définie par :

$$\Im(S_{i+1}) = I(S_i) - I(S_{i+1}) \qquad (3.\ 8)$$

Ce gain d'incertitude, appelé aussi gain informationnel, représente une variation d'entropie, et plus spécifiquement l'entropie de Shannon [152].

Entropie de Shannon: soit (p_1 , p_2 ,..., p_m) un élément du simplexe Σ_m, il s'agit généralement d'un vecteur de probabilité. On appelle entropie de Shannon l'application h_s définie par:

$$h_s : \Sigma_m \to R_+$$

$$h_s(p_1, p_2, ..., p_m) = -\sum_{i=1}^{m} p_i \log_2 p_i \qquad (3.\ 9)$$

En pratique, comme on travaille sur des échantillons de taille finie, ces probabilités sont estimées par les fréquences empiriques. Ce qui donne pour un sommet s_k :

$$h_s(s_k) = -\sum_{i=1}^{m} \frac{n_{ik}}{n_k} \log_2 \frac{n_{ik}}{n_k} \qquad (3.\ 10)$$

où n_{ik} représente les effectifs du sommet s_k qui appartiennent à la classe c_i, $n_{.k}$ les effectifs du sommet s_k, n_i les effectifs de la classe i dans la partition Ω_a, n les effectifs de Ω_a et *m* le nombre de classes. Ainsi:

$$n_{.k} = \sum_{i=1}^{m} n_{ik} \qquad (3.\ 11)$$

$$n_i = \sum_{k=1}^{K} n_{ik} \qquad (3.\ 12)$$

$$n = \sum_{k=1}^{K} n_{.k} \qquad (3.\ 13)$$

Dans le cas d'une partition S_K à *K* éléments, l'incertitude est l'entropie conditionnelle moyenne des sommets de cette partition exprimée par :

$$I(S_K) = \sum_{k=1}^{K} \frac{n_k}{n} \left(-\sum_{i=1}^{m} \frac{n_{ik}}{n_k} \log_2 \frac{n_{ik}}{n_k}\right) \qquad (3.\ 14)$$

La condition d'arrêt pour l'algorithme ID3 peut se faire selon différentes options:

a. Gain d'information minimal : On fixe un seuil β en dessous duquel on refuse d'éclater un sommet. Ainsi, un sommet est déclarée saturée s'il n'y a aucune variable X_j capable d'engendrer une nouvelle partition S_{i+1} telle que $\Im(S_{i+1}) \geq \beta$.

b. Test statistique : Dans ID3, il est possible d'arrêter le processus de segmentation en utilisant un test d'indépendance du χ^2. Le principe pour un sommet est le suivant. Les seules variables candidates à la segmentation sont celles qui permettent de construire un tableau de contingence dont le χ^2 est supérieur à une valeur critique η_α fixée par l'utilisateur. Concrètement:
Soit l'hypothèse H_0 : X_j et C sont indépendants
Si H_0 est vérifiée, il n'y a donc aucun intérêt à retenir la variable X_j pour partitionner le sommet.

Sous H_0, la quantité :

$$\chi^2_{calcul} = \sum_{i=1}^{m} \sum_{k=1}^{\alpha_j} \frac{(n_{ij} - n_{i.} n_j)^2}{n_i n_j} \qquad (3.\ 15)$$

suit une loi du χ^2 à $(\alpha_j\text{-}1)\times(\text{m-}1)$ degrés de liberté.(α_j est le cardinal de la partition du sommet). On fixe ensuite le risque de première espèce α. On peut alors déterminer grâce à une table du χ^2 à $(\alpha_j\text{-}1)\times(\text{m-}1)$ degrés de liberté la valeur η_α délimitant l'intervalle de rejet et appliquer la règle suivante:

$$\text{si } \chi^2_{calcul} \geq \eta_\alpha \text{, alors on rejette } H_0 .$$

Si aucune variable n'est retenue, le sommet est déclaré saturé.

3.7.3 Algorithme C4.5

C4.5 est une extension de ID3. La différence entre les deux algorithmes se présente sur les points suivants :

- Dans la construction de l'arbre de décision, il est possible de classifier les enregistrements ayant des valeurs inconnues en estimant la probabilité des différents résultats possibles.

- C4.5 permet la discrétisation de valeurs continues des attributs, en faisant des ensembles ou de plages selon la technique suivante. Soit un attribut X_i continu. Toutes les valeurs de cet attribut dans l'ensemble d'apprentissage sont classées dans un ordre croissant et ensuite, pour chacune des valeurs A_j on partitionne les enregistrements en deux catégories : ceux qui ont une valeur pour X_i inférieur ou égale à A_j et ceux qui ont une valeur pour X_i supérieur à A_j. Pour chacune de ces partitions, on calcule le ratio de gain, et on choisit la partition qui maximise ce dernier.

- Dans cet algorithme, Quinlan a introduit dans le critère de partitionnement un facteur visant à pénaliser la prolifération des sommets. Il désavantage donc les variables qui ont beaucoup de modalités et évite un émiettement de la population. Le critère obtenu s'appelle le gain ratio et qui représente le rapport entre le gain d'incertitude et la distribution des individus sur la partition S=$\{s_1, s_2, ..., s_\alpha\}$ produite suite à une segmentation d'un sommet par une variable X_j.

$$\Im(S_{i+1}) = \frac{I(S_i) - I(S_{i+1})}{-\sum_{k=1}^{\alpha} \frac{n_k}{n_j} \log_2 \frac{n_k}{n_j}} \qquad (3.16)$$

- Une autre caractéristique de C4.5 est qu'il procède à un élagage statistique de l'arbre. La démarche consiste à produire un arbre développé au maximum et ensuite à supprimer les sous-arbres ne vérifiant pas une certaine condition reposant sur le taux d'erreur. Le principe consiste à supprimer les partitions dont le taux d'erreur moyen est supérieur à celui du sommet père.

- Dans C4.5 on peut fixer une contrainte d'admissibilité, Cette contrainte consiste à rejeter une variable si elle génère une partition dans laquelle un nombre ρ de sommets possède un effectif inférieur à une valeur τ. On dit qu'une telle partition n'est pas admissible. Si, pour un sommet donné, toutes les variables conduisent à des partitions non admissibles, le sommet est déclaré saturé.

Sur l'importance de cet algorithme Kodratoff [132] a dit « Dans notre expérience, nous n'avons jamais rencontré un cas où C4.5 soit surclassé par un réseau neuronal, ce qui est largement confirmé dans la littérature d'apprentissage automatique ».

3.7.4 Algorithme SIPINA

L'algorithme SIPINA [127] fournit une suite de partitions non nécessairement hiérarchisées. C'est un graphe d'induction non arborescent, il tente de réduire les inconvénients des méthodes arborescentes d'une part par l'introduction de l'opération de fusion et d'autre part par l'utilisation d'une mesure sensible aux effectifs.

L'algorithme de construction est une heuristique qui produit une succession de partitions par fusion et /ou éclatement des nœuds du graphe. Le passage d'une partition à une autre est obtenue en maximisant la variation d'incertitude $\Im_\lambda$ entre la partition courante et la partition précédente qui s'exprime par :

$$\Im_\lambda(S_{i+1}) = I_\lambda(S_i) - I_\lambda(S_{i+1}) \qquad (3.17)$$

avec $I_\lambda(S_i)$ la mesure d'entropie de la partition S_i et $I_\lambda(S_{i+1})$ la mesure d'entropie de la partition suivante S_{i+1}. Pour $I_\lambda(S_i)$ nous pouvons utiliser plusieurs fonctions construites à partir de mesure d'incertitude telles que :

- *l'entropie de Shannon :*

$$I_\lambda(S_i) = \sum_{j=1}^{K} \frac{n_j}{n} \left(- \sum_{i=1}^{m} \frac{n_{ij} + \lambda}{n_j + m\lambda} \log_2 \frac{n_{ij} + \lambda}{n_j + m\lambda} \right) \quad (3.\ 18)$$

- *l'entropie quadratique :*

$$I_\lambda(S_i) = \sum_{j=1}^{K} \frac{n_j}{n} \left(- \sum_{i=1}^{m} \frac{n_{ij} + \lambda}{n_j + m\lambda} \left(1 - \frac{n_{ij} + \lambda}{n_i + m\lambda}\right) \right) \quad (3.\ 19)$$

n_{ij} : le nombre d'individus de la classe i se trouvant au sommet S_j

n_i : le nombre total d'individus de la classe i dans une partition, $n_i = \sum_{j=1}^{k} n_{ij}$

n_j : le nombre d'individus d'un sommet S_j, $n_j = \sum_{i=1}^{2} n_{ij}$

n : le nombre total d'individus, $n = \sum_{i=1}^{k} n_i$

m : le nombre de classes.

λ : contrôle le développement du graphe et pénalise les nœuds de faibles effectifs et, de ce fait, favorise les fusions entre les sommets semblables. L'une des stratégies utilisées pour fixer λ consiste à définir une situation indésirable et de trouver λ^* qui pénalise le plus cette situation.

3.7.4.1 *Fixation du paramètre λ*

Le paramètre λ contrôle le développement du graphe en pénalisant les nœuds qui ont des effectifs trop faibles. Il s'agit donc d'un paramètre qui va contrôler et favoriser la fusion entre les sommets, cette flexibilité étant une spécificité majeure de la méthode Sipina. La méthode adoptée pour trouver ce λ consiste à définir une situation indésirable et trouver la valeur de λ que nous noterons λ^* qui pénalise le plus cette situation.

Nous considérons aussi le paramètre d'admissibilité qui fixe le nombre en dessous duquel les effectifs d'un sommet sont jugés trop faible pour que le sommet soit pris en compte. Nous fixons par exemple ce paramètre d'admissibilité à τ. Prenons un exemple simple de deux distributions possibles. Nous allons prendre deux colonnes T_u et T_v du tableau de contingence. Le principe est que si un nœud présente une distribution dont les effectifs totaux sont inférieurs ou égaux à τ, la valeur du critère utilisé devrait être la plus mauvaise possible.

T_u	T_v
τ	τ+1
0	0
⋮	⋮
0	0
$n_{.i} = \tau$	$n_{.j} = \tau + 1$

Figure 3. 6. *Exemple de deux distributions*

A priori, on tend à penser que les deux distributions sont les mêmes surtout qu'elles ne différent en tout et pour tout que d'une unité. Cependant, par rapport au critère d'admissibilité choisi, nous devrions pénaliser et ce de manière assez forte la distribution T_u puisqu'elle a des effectifs que l'on pourrait qualifier de limités. Nous choisirons λ* telle que :

$$\lambda^* \text{ tel que } I_{\lambda^*}(T_u) - I_{\lambda^*}(T_v) = \max_{\lambda}\left(I_{\lambda}(T_u) - I_{\lambda}(T_v)\right) \qquad (3.\ 20)$$

avec

$$I_{\lambda}(T_j) = \left(\sum_{i=1}^{m} \frac{n_{ij} + \lambda}{n_{.j} + m\lambda}\left(1 - \frac{n_{ij} + \lambda}{n_{.j} + m\lambda}\right)\right) \qquad (3.\ 21)$$

Ce qui nous donne, finalement:

$$I_{\lambda^*}(T_v) - I_{\lambda^*}(T_u) = \lambda(m-1)\frac{2\tau^2 + 2\tau + m\lambda + 2m\lambda\tau}{(\tau + m\lambda)^2(\tau + 1 + m\lambda)^2} \qquad (3.\ 22)$$

que l'on doit maximiser.

3.7.4.2 *Fixation de la contrainte d'admissibilité*

Il existe deux stratégies pour déterminer la taille minimale d'un sommet. La première consiste à demander à l'utilisateur de fixer le nombre minimum d'individu τ que doit comporter chaque sommet. La seconde consiste à calculer le nombre minimum en adoptant un point de vue statistique que nous décrivons ci-dessous.

Dans le cadre de cette thèse, nous cherchons seulement à discriminer deux classes c_1 et c_2. Les éléments de la classe c_1 seront appelés les *exemples* et ceux de *c2* les *contre-exemples*. Le raisonnement étant parfaitement symétrique si on inversait les deux classes. Dans la suite, on notera la classe des exemples *e* et celle des contre-exemples *c*. Le nombre d'individus dans la classe exemple sera n_e et le nombre d'individus dans la classe des contres exemples sera n_c. La taille de l'échantillon d'apprentissage est donc $n=n_e+n_c$. Soit *s* un sommet terminal du graphe d'induction dont les effectifs totaux sont $n_s=n_{se}+n_{sc}$ où n_{se} et n_{sc} désignent respectivement le nombre d'individus de la classe des exemples et le nombre d'individus de la classe des contre-exemples dans le sommet *s*. Soit le test d'hypothèse suivant

$$\begin{cases} H_0 : \dfrac{n_{sc}}{n_s} = \dfrac{n_c}{n} \\ H_1 : \dfrac{n_{sc}}{n_s} < \dfrac{n_c}{n} \end{cases} \quad (3.\ 23)$$

H_0 signifie que la règle qui conduit au sommet s n'est pas pertinente car la proportion de contre exemples dans ce sommet est identique à ce qu'elle était à la racine. Autrement dit, la règle *R* qui définit le sommet *s* ne nous apprend rien sur la classe *e* des exemples. Sous H_0 Lerman [151] a montré que le nombre de contre-exemple n_{sc} suit une loi de Poisson de paramètre :

$$\theta = \frac{n_c \times n_s}{n} \quad (3.\ 24)$$

La région critique du test s'écrit :

$$\sum_{x=0}^{n_{sc}} e^{-\lambda} \frac{\theta}{x!} \leq \alpha_0 \quad (3.\ 25)$$

Nous fixons l'effectif minimum τ comme suit : pour un seuil critique α_0=5% par exemple, on détermine le nombre d'individus total dans un sommet terminal n_s tel que si n_{sc}=0, on conclut au rejet de H_0. Ainsi, on établit :

$$e^{-\theta} = \alpha_0 \quad (3.\ 26)$$

d'où on extrait :

$$n_s = -\ln(\alpha_0)\frac{n}{n_c} \quad (3.\ 27)$$

la valeur de $\tau = n_s$

La détermination de τ ne pose donc aucun problème puisque le risque critique α_0 est fixé par l'utilisateur et les effectifs *n* et n_c sont connus dès le départ.

3.8 Evaluations des classifieurs

Notre objectif est de trouver le meilleur classifieur qui soit le mieux adapté à notre problème. En effet, il est considéré qu'il n' y a pas une méthode d'apprentissage qui surpasse les autres, car chacune des méthodes à ses forces et ses faiblesses. Il est recommandé de tester plusieurs méthodes afin de trouver la meilleure d'entres elles ou de les faire coopérer.

La sélection des modèles est l'un des champs les plus lucratifs de l'analyse expérimentale [137] [138], elle répond objectivement à la question « quelle est la meilleure méthode sur les données compte tenu de notre objectif ? ».

Il est délicat de formuler des indicateurs généraux pour valider les modèles, dans la plupart des cas, les chercheurs travaillent sur le taux d'erreur parce qu'il est le seul indicateur qui soit véritablement comparable d'un algorithme à une autre, mais bien sûr on trouve aussi d'autres critères comme la complexité et le temps de réponse.

En général, la performance d'un modèle s'apprécie au travers d'une matrice de confusion, qui compare la situation réelle et la situation prévue par le modèle afin d'estimer le taux d'erreur.

3.8.1 Matrice de confusion

La matrice de confusion présente sous la forme d'un tableau de contingence confrontant la classe d'affectation (en colonne) avec la classe d'origine (en ligne) des individus composant l'échantillon. Nous disposons de deux types d'informations :

- Le nombre de fois où le modèle s'est trompé
- Le type d'erreur lors du classement

La figure 3.7 présente une matrice de confusion pour un modèle de 2 classes *A* et *B*.

	Classe d'affectation	
Classe d'origine	A	B
A	$n_{A.A}$	$n_{A.B}$
B	$n_{B.A}$	$n_{B.B}$

Figure 3. 7. *Matrice de confusion pour 2 classe A et B*

Dans cette matrice, $n_{A.B}$ représente le nombre de cas de la classe *A* affectés à la classe *B* et $n_{B.A}$ représente le nombre de cas de la classe *B* affectés à la classe *A*, alors que $n_{A.A}$ et $n_{B.B}$ représentent le nombre correct de classification.

A partir de cette matrice de confusion on peut dégager trois types d'indicateurs :

1. le taux d'erreur globale en resubstitution : Ce taux d'erreur est calculé sur l'échantillon d'apprentissage Ω_a, il est généralement optimiste, c'est à dire plus faible que le taux d'erreur théorique qui représente la probabilité que l'on se trompe si on applique le classifieur sur toute la population Ω, ce qui est impossible. Afin de palier ce biais d'optimisme, très difficile à estimer, on propose généralement de subdiviser les échantillons qu'on a en deux parties, une base de données d'apprentissage Ω_a et une base de test Ω_t. Ces deux bases comprennent des enregistrements différents. Par expérience, la base d'apprentissage reprendra de 70% à 80% des enregistrements, la base test étant constituée des 20 à 30 restants. La base d'apprentissage sert donc à construire le modèle et la base test sert à vérifier la stabilité du modèle. Dans ce cas, nous calculons le taux d'erreur qu'on appelle taux d'erreur globale en validation. Pour la matrice ci-dessus le taux d'erreur globale est calculé comme suit :

$$\varepsilon_{globale} = 1 - Succès \qquad (3.\ 28)$$

$$\varepsilon_{globale} = 1 - \frac{n_{A.A} + n_{B.B}}{card(M)} \quad (3.\ 29)$$

où *card*(*M)* est le nombre total d'individu

2. le taux d'erreur *à priori* : c'est la probabilité qu'un individu appartenant à la classe *k* ne soit pas affecté à la classe *k*. Pour notre exemple et pour la classe *A*, le taux d'erreur à priori est donné par l'équation suivante :

$$\varepsilon_{à\ priori} = \frac{\sum_{k \neq A} n_{A,k}}{\sum_{k} n_{A,k}} \quad (3.\ 30)$$

avec *k* représente les différentes classes, dans notre cas *A* ou *B*

3. le taux d'erreur *a posteriori* : c'est la probabilité qu'un individu affecté à la classe *k* appartienne effectivement à la classe *k.* pour notre exemple et pour la classe *A*, le taux d'erreur a posteriori est :

$$\varepsilon_{à\ posteriori} = \frac{\sum_{k \neq A} n_{k.A}}{\sum_{k} n_{k.A}} \quad (3.\ 31)$$

Le taux d'erreur global nous permet de savoir comment va agir un classifieur sur l'ensemble des données ; cependant, il ne nous permet pas de distinguer quel est le niveau de réussite pour chaque classe. C'est pourquoi le taux d'erreur à priori a été calculé. Il s'agit en effet de calculer le taux de réussite relatif à chaque classe. Il s'agit du complément du critère classique du taux de rappel utilisé le plus souvent dans les systèmes de recherche d'information (SRI). Le taux d'erreur à posteriori nous permet de mettre l'accent sur la crédibilité d'un classement, comme par exemple, savoir la certitude qu'un individu classé A soit bien de cette classe A. Il est donc le complément du classique taux de précision tel que l'on utilise dans les SRI.

L'évaluation des performances partielles permet de déterminer sur quelles erreurs le classifieur est moins performant, de comparer plusieurs classifieurs, de les faire coopérer en utilisant sur les catégories qu'ils prédisent mieux.

3.8.2 Validation croisée

La validation croisée ou cross-validation propose de diviser la base d'échantillon en "*s*" partie égales, avec apprentissage sur les (*s*-1) parties de la base, et test sur la partie qui reste. Ensuite, il y a une permutation des bases testées, et donc un tableau de confusion est crée en faisant la moyenne des "*s*" tests effectués. Il s'agit donc d'une répétition du couple « apprentissage-validation », mais en veillant à ce qu'il n'y ait aucun recouvrement entre les échantillons de validation [139].

Algorithme 3.3 : Algorithme Cross-validation

Subdiviser l'échantillon de départ en *S* parties égales de *n* individus chacune
Initialiser : portion courante;
répéter
  Former l'échantillon $\Omega_a = \Omega - n$
  Créer un classifieur $\varphi_{\Omega-n}$
  Calculer l'erreur ε_n
jusqu'à fin de toutes les portions

$$\varepsilon = \sum_{S_i} \frac{card(\Omega_a, n)}{card(\Omega_a)} \varepsilon_n \quad \text{et la variance est: } Var(\varepsilon) \approx \frac{\varepsilon(1-\varepsilon)}{S}$$

3.8.3 Le Bootstrap

L'idée est d'utiliser l'échantillon des observations pour permettre une inférence statistique plus fine. On réalise un certain nombre d'échantillons, qualifiés d'échantillon bootstrap, obtenus par tirage aléatoire d'observations de l'échantillon initial. Sur chacun des échantillons bootstrap, on estime les différents paramètres du modèle. On obtient par conséquent une suite de paramètres. Sous certaines conditions de régularité, la théorie montre que la distribution de la suite de paramètres obtenus converge vers la réelle distribution du paramètre.

Le bootstrap s'est aujourd'hui imposé dans le domaine statistique comme une technique très pratique d'inférence statistique. Elle nécessite, en effet, « peu » d'hypothèses et est relativement facile à programmer, ce ne sont, en effet, que des tirages aléatoires. Toutefois, le gros inconvénient réside dans les importantes capacités de calcul que l'application de ces techniques exige, en plus il est souvent recommandé de procéder à au moins une centaine de répétitions pour espérer avoir une bonne fiabilité

3.9 Conclusion

Dans nos travaux, nous avons opté pour les graphes d'induction pour l'apprentissage d'un classifieur. L'induction par graphes, représente l'avantage d'utiliser un système de représentation remplissant parfaitement les critères de précision et surtout de compréhensibilité. Leur simplicité conduit à des algorithmes qui ne sont pas trop compliqués et qui requièrent peu de calculs, donc particulièrement adaptés pour le traitement de grosses bases de données. Leur mise à jour ne pose pas de problèmes particuliers et en plus ils permettent de sélectionner des meilleurs attributs prédictifs. Enfin, leur rapidité en classement dépend tout simplement du nombre moyen de nœud en partant de la racine aux feuilles.

Un consensus général semble se dégager pour reconnaître qu'aucune méthode basée sur les graphes ne surpasse les autres car elles ont toutes leurs forces et faiblesses spécifiques les rendant ainsi plus ou moins performantes pour un problème donné. Dans certains cas, il est plus avantageux de faire coopérer des méthodes comme nous avons fait pour le problème de classification de sites adultes que nous détaillerons dans le chapitre 5.

Le chapitre suivant (chapitre 4) présentera le processus complet de la construction de notre modèle de peau dans une démarche d'extraction de connaissances à partir de données issus d'images.

Chapitre 4

Modèle de peau

MODELE DE PEAU 67

4.1 Introduction

Dans le chapitre précédent nous avons introduit le processus complet d'extraction de connaissances à partir des données. Dans ce chapitre, nous nous intéressons au problème de découverte de connaissances à partir des données images et plus particulièrement d'un nombre important de pixels. Notre objectif est de construire un modèle de prédiction permettant d'identifier les pixels de couleur de peau dans une image, en utilisant l'indice couleur qui est la primitive la plus simple à calculer et la plus riche. Pour atteindre cet objectif, un système d'apprentissage supervisé doit alors, à partir d'un ensemble d'exemples de pixels, extraire une procédure de classification qui doit déterminer si un pixel est un pixel de peau ou non.

L'approche adoptée trouve son originalité dans le choix des axes de couleur les plus pertinents, la manière dont ce choix est fixé, et l'utilisation de la distribution spectrale dans le processus d'extraction des régions de peau.

Nous comparons dans ce chapitre les résultats obtenus à partir de notre modèle avec celui de Compaq et nous évaluons sa performance à travers différentes applications :

- La détection de visage, un système élaboré au sein de notre équipe
- La classification de portraits en gros plan, plan américain et plan en pied
- Le filtrage de sites pornographiques. Cette partie constitue l'application principale de notre travail et nous la détaillerons dans le chapitre suivant.

4.2 Notre démarche

Pour l'élaboration du modèle de peau, on a procédé en trois étapes (cf. figure 4.1) :

- La première étape est consacrée à la constitution de notre corpus, ainsi qu'à la préparation de données pour la phase d'apprentissage.

- La seconde étape consiste à trouver un modèle de prédiction associé à un espace de représentation capable de discriminer les pixels de peau de ceux de non-peau. Cet espace de représentation peut être un des espaces de couleur classiques ou un nouveau type d'espace hybride en sélectionnant un ensemble de composantes issues de différents espaces de couleur classiques. Les outils de data mining nous permettent ensuite de retenir les axes les plus pertinents, ainsi que les règles de décision adéquates. Nous montrons, dans cette étape, l'importance de la distribution spectrale dans le processus de classification.

- La dernière étape est celle de la segmentation de l'image et l'identification des différentes zones d'intérêt (de peau) en tenant compte des règles extraites lors de la phase précédente.

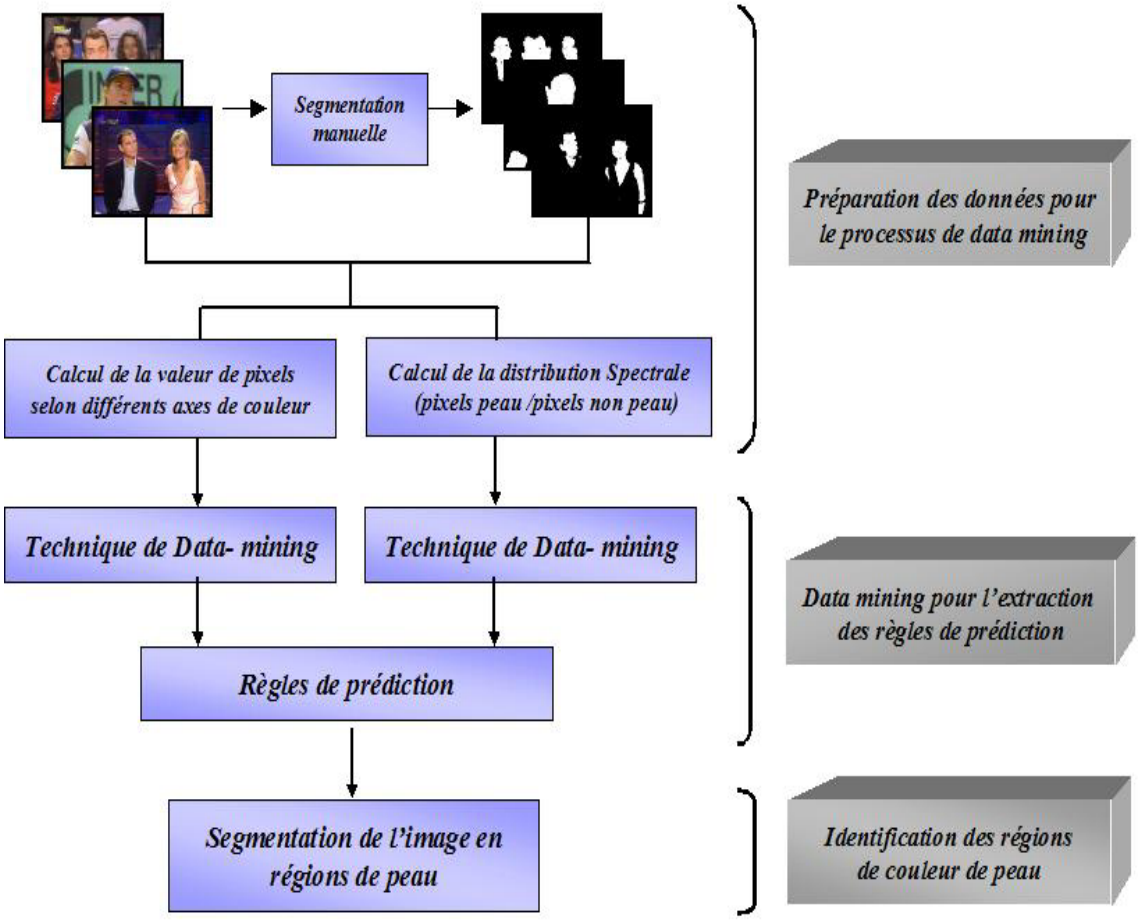


Figure 4. 1. *Démarche adoptée*

4.3 Description du corpus

La construction de la base d'apprentissage est un élément important dans une démarche d'extraction de connaissances à partir des données. Pour notre problème de classification de pixels en pixels de peau ou non, la qualité du corpus peut être jugée sur les facteurs suivants :

1. la taille de la base de données et la variété dans les contenus d'images, qui doit être représentative pour les différents sexes, races, et conditions d'éclairages.
2. la source de la base d'images, qui dépend particulièrement de l'application, tels que le filtrage de sites adultes sur Internet, la détection de visage dans la vidéo, etc.

Nous utilisons un corpus large d'images, contrairement à la majorité des travaux existants, qui pratiquent une phase d'apprentissage sur des classes prédéfinies d'images, sous des conditions d'éclairage connues à l'avance. Si ces modèles conviennent généralement à des systèmes à base d'images spécifiques donc peu variées par exemple aux conditions d'éclairage, ils sont peu adaptés aux systèmes contenant une grande variété d'images.

Nous travaillons sur une base d'images composée du corpus CRL de Compaq et d'une base développée au sein de notre équipe nommée ECL SCIV (ECL Skin-Color Images from

Video). Le corpus CRL[70] résulte d'une base de 12 230 images collectées par un Web crawler, conduisant à un modèle général composé de 1.949.695.888 pixels dont 80 377 671 pixels de peau et de 854 744 181 pixels de non peau. La base ECL SCIV[172]est composée de plus de 1110 images de couleurs de peau de différentes personnes issues de TV (Euronews, TF1, France 2) couvrant des programmes divers et variés tels que les journaux, des films, du sport, etc. Elle conduit à un modèle général composé de 85 248 000 pixels correspondant à plus de 30 heures de vidéo. Cette base a été nécessaire pour couvrir les conditions d'éclairage d'images vidéo que l'on trouve fréquemment dans les sites adultes et où l'on applique très souvent une correction gamma sur les couleurs de façon à ce que les visages n'apparaissent pas trop pales à l'image. La figure 4.2 montre un extrait de cette base d'images ECL SCIV.

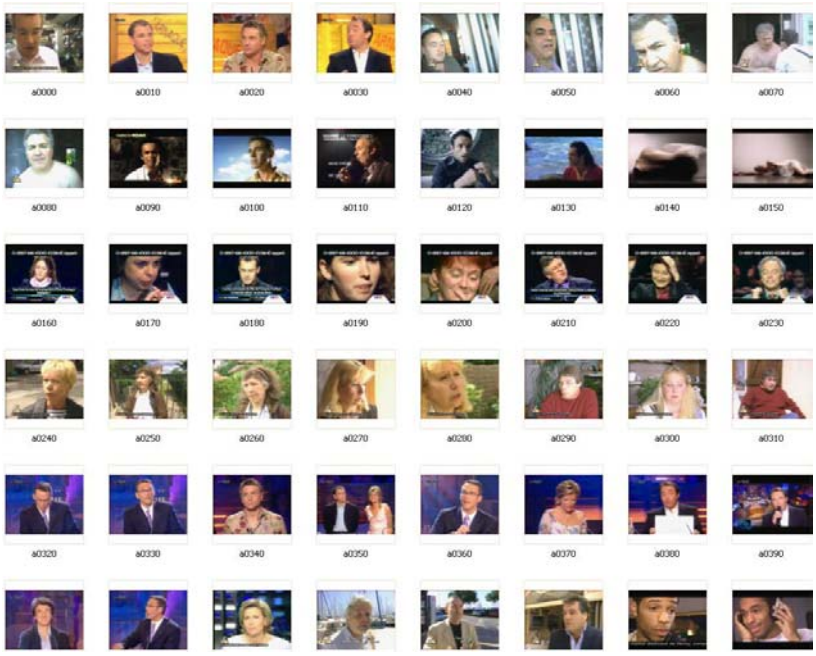


Figure 4. 2. *Extrait de la base de données d'ECL SCIV.*

La base ECL a aussi la particularité de contenir, en plus des masques binaires (cf. figure 4.3), des méta-données sur le contenu de chaque image tels que le sexe, la race, la prise de vue (intérieur/extérieur), jour et nuit.

Figure 4. 3. *Image (gauche) et son masque binaire (droite)*

4.4 Espaces de couleur étudiés

Dans la mesure où la couleur de peau est la perception de la lumière réfléchie par une surface de peau dans une image, nous proposons une classification des pixels dont les couleurs sont représentées dans un espace de couleur qui permet la meilleure discrimination possible entre les classes de pixels de peau et de non peau. De ce fait, nous présentons dans cette section les différents espaces de couleur que nous avons étudiés pour l'élaboration de notre modèle.

Le choix de l'espace de couleur est très important pour la perception des couleurs proches pour l'utilisateur. Les images sont souvent représentées en espace RGB. Suivant les applications, les caractéristiques sont plus perceptibles dans certains espaces plutôt que dans d'autres.

Il existe plusieurs systèmes de représentation de la couleur qu'on peut classer en quatre familles [173] :

- Les systèmes de primaires, tels que le système (X,Y,Z) de la C.I.E. et les systèmes (R,G,B) et (r,g,b)
- Les systèmes luminance-chrominance, dans lesquels une composante représente la luminance et deux composantes la chrominance d'un stimulus de couleur. Nous distinguons différents types de ces systèmes :
    - Les systèmes perceptuellement uniformes, tels que Lab et Luv.
    - Les systèmes de télévision, tels que YIQ et YUV, qui permettent de séparer l'information de chrominance de l'information de luminance pour la transmission de signaux de télévision.
- Les systèmes perceptuels, qui représentent la couleur selon des entités telles que la luminance, la teinte et la saturation. Nous distinguons deux familles :
    - Les systèmes de coordonnées polaires ou cylindriques, tels que HSV.
    - Les systèmes humains de la perception de la couleur. Ils sont évalués directement à partir des composantes trichromatiques d'un système de primaires et se différencient par les relations exprimant la luminosité, la teinte ou la saturation.
- Les systèmes d'axes indépendants.

La figure 4.4 illustre la représentation d'une image dans quelques espaces de couleur.

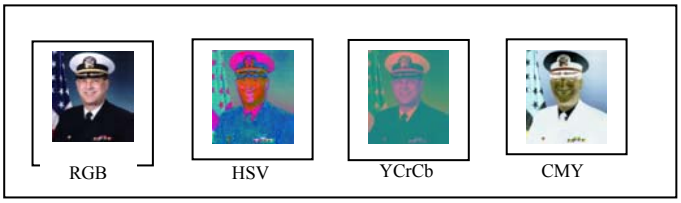


Figure 4. 4. *Représentation d'une image dans différents espaces de couleur*

Nous présentons par la suite les espaces de couleur que nous avons utilisés :

- ***Le modèle RGB***

La Commission Internationale de l'Eclairage (CIE) a choisi en 1931 les trois longueurs d'onde suivantes pour représenter les trois couleurs fondamentales :

* Bleu 435,8 nm * Vert 546,1 nm * Rouge 700 nm

Dans un tel modèle, les trois axes correspondent aux couleurs primaires Rouge, Vert, Bleu (figure 4.5).

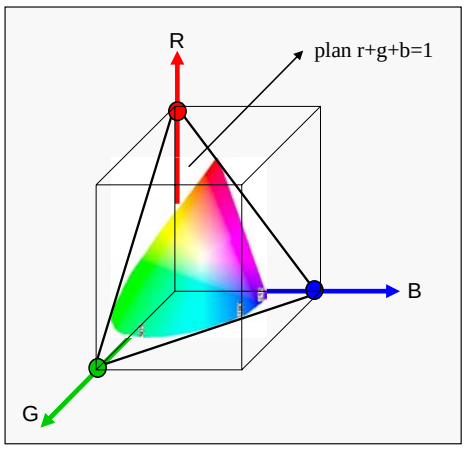


Figure 4. 5. *Le système RGB*

La diagonale principale représente les niveaux de gris. Ce modèle constitue le principe de base des moniteurs de télévision et des écrans à balayage ; en effet, c'est par superposition de rouge, de vert et de bleu que l'affichage couleur est réalisé.

Le système RGB normalisé correspond aux chromacités en RGB.

$$r = \frac{R}{R+G+B} ;\ g = \frac{G}{R+G+B} ;\ b = \frac{B}{R+G+B} \qquad (4.\ 1)$$

- ***Le modèle YCrCb***

L'espace de couleur YCrCb est utilisé dans le standard JPEG.

$$\begin{pmatrix} Y \\ Cr \\ Cb \end{pmatrix} = \begin{vmatrix} 0.2990 & 0.5870 & 0.1140 \\ 0.5000 & -0.4187 & -0.0813 \\ -0.1687 & -0.3313 & 0.5000 \end{vmatrix} \begin{pmatrix} R \\ V \\ B \end{pmatrix} \qquad (4.\ 2)$$

- ***Le modèle YIQ***

Il s'agit d'un recodage du système RGB par NTSC (National Television Standards Committee). Les relations entre ces paramètres et le modèle RGB sont les suivantes :

$$\begin{pmatrix} Y \\ I \\ Q \end{pmatrix} = \begin{vmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{vmatrix} \begin{pmatrix} R \\ V \\ B \end{pmatrix} \quad (4.3)$$

Où Y est la luminance et I et Q représentent respectivement la teinte et la saturation. C'est un modèle qui est fondé sur des observations psychophysiques [174].

- ***Le modèle HSV***

Le modèle Teinte-Saturation-Luminance ou HSV (Hue, Saturation, Value) est plus proche de la perception de la couleur, ce modèle utilise un espace en forme d'hexagone dont l'axe est celui de la luminance L. Pour L = 1, on a les couleurs d'intensité maximale.

La teinte T est donnée par l'angle entre l'axe rouge et un point de l'hexagone. La saturation S est donnée par la distance entre l'axe de la luminance et un point de l'hexagone. L'espace de couleur TSL est de forme cylindrique (figure 4.6).

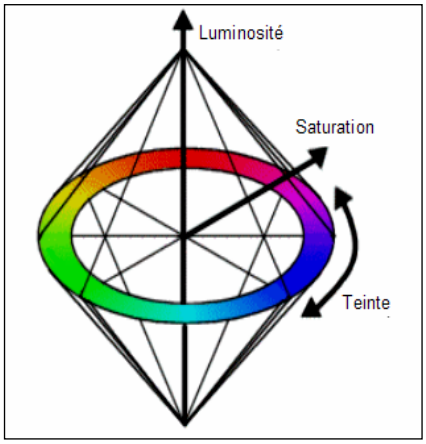


Figure 4. 6. *Espace de couleur HSV*

La séparation d'une couleur en différentes composantes de teinte, saturation et luminance sont intuitives pour l'utilisateur.

$$H = \begin{cases} \dfrac{60.(g-b)}{\max(r,g,b)-\min(r,g,b)} & pour\ r = max(r,g,b) \\ 120 + \dfrac{60.(g-b)}{\max(r,g,b)-\min(r,g,b)} & pour\ g = \max(r,g,b) \\ 240 + \dfrac{60.(g-b)}{\max(r,g,b)-\min(r,g,b)} & pour\ b = max(r,g,b) \end{cases} \quad (4.4)$$

$$S = \begin{cases} \dfrac{\max(r,g,b)-\min(r,g,b)}{\max(r,g,b)+\min(r,g,b)} & pour\ 0 < L \le 0.5 \\ \dfrac{\max(r,g,b)-\min(r,g,b)}{20-\max(r,g,b)+\min(r,g,b)} & pour\ 0.5 \le L < 1.0 \end{cases} \quad (4.5)$$

$$V = \max(r,g,b) \quad (4.6)$$

Si $H < 0$ alors $H = 360 + H$

- ***Le modèle CMY***

Le codage CMY (Cyan, Magenta, Yellow, ou Cyan, Magenta, Jaune en français, soit CMJ) est à la synthèse soustractive, ce que le codage RGB est à la synthèse additive. Ce modèle consiste à décomposer une couleur en valeurs de Cyan, de Magenta et de Jaune.

4.5 Identification des pixels de peau

Dans cette partie, nous détaillons les différentes phases de construction de notre modèle de peau, en nous appuyant sur des techniques de data mining et d'analyse d'images. La méthode utilise les techniques de data mining pour produire les règles de prédiction en fonction des espaces de couleur, suivie d'une phase d'identification et de segmentation en régions cohérentes de peau en utilisant les règles déjà produites.

4.5.1 Construction de l'espace couleur hybride adapté

De nombreux auteurs ont tenté de déterminer les espaces de couleur qui sont les mieux appropriés pour discriminer les pixels de peau et non peau. Nous avons vu au chapitre 2 qu'ils fournissent des réponses contradictoires sur la pertinence des espaces de couleur. Notre objectif est de définir, par des techniques de data mining, l'espace de couleur le plus discriminant. Nous proposons de classer les pixels de peau/non peau dans différents espaces de représentation, par une approche supervisée, afin de déduire des règles de décision pertinentes. Dans la suite, nous développons d'abord l'approche bayésienne, qui repose donc sur des probabilités de pixels à la classe peau ou non peau ; c'est ce que nous avons exploré en premier lieu. Nous avons aussi exploré une exploitation directe des composantes couleurs de pixels suite à des difficultés constatées lors des études précédentes.

4.5.1.1 *Approche bayésienne*

Nous avons démarré notre étude par une approche bayésienne, l'objectif étant d'améliorer l'approche utilisée par Compaq [80]. Nous cherchons donc à identifier un pixel de peau avec un degré élevé de précision en utilisant des règles de prédiction en fonction de différents espaces de couleur, contrairement à Compaq qui utilise uniquement l'espace RGB avec un seuil fixé arbitrairement.

Afin de réduire la complexité de l'étude, on s'est borné à l'utilisation de deux axes pour la caractérisation de la couleur de peau, notre intuition étant que deux axes suffisent à discriminer les couleurs de peau de celles de non peau. Conformément à une approche bayésienne, nous avons donc construit des histogrammes de couleur de peau et de non-peau selon différentes combinaisons d'axes de couleur afin de déduire par la suite les combinaisons pertinentes qui représentent le mieux la distribution des couleurs de pixels de peau [3].

La figure 4.7 illustre les 12 composantes que nous avons choisies pour la construction de différents histogrammes de couleur qui sont dans l'ordre:

R	G	B	H	S	V	Y	CB	CR	I	R-G	H1

Figure 4. 7. *Les composantes d'espaces de couleur* utilisées

A partir de 12 composantes nous pouvons donc construire 78 combinaisons d'axes. Le tableau 4.1 montre les différentes combinaisons possibles.

Tableau 4. 1. *Les 78 combinaisons* d'axes

	R	G	B	I	R-G	H1	H	S	V	Y	CB	CR
R	1	2	3	4	5	6	7	8	9	10	11	12
G		13	14	15	16	17	18	19	20	21	22	23
B			24	25	26	27	28	29	30	31	32	33
I				34	35	36	37	38	39	40	41	42
R-G					43	44	45	46	47	48	49	50
H1						51	52	53	54	55	56	57
H							58	59	60	61	62	63
S								64	65	66	67	68
V									69	70	71	72
Y										73	74	75
CB											76	77
CR												78

Pour construire les histogrammes h_i (avec i=1 jusqu'à 78) de couleur de peau et de non-peau, nous avons associé pour chaque case (bins) des axes utilisés, le nombre de fois que la valeur de couleur s'est produite dans la base de données des images. Les pixels de 12230 images du corpus CRL ont été utilisés pour peupler l'histogramme. Les pixels de peau dans les 3265 images contenant la peau (marqués manuellement), sont placés dans l'histogramme de peau, alors que les pixels de 8965 images ne contenant pas de peau sont placés dans l'histogramme non-peau.

L'algorithme permettant de calculer l'histogramme h_i de l'échantillon de pixel E à partir des deux axes C_1 *et* C_2 est donné par l'algorithme 4.1.

Algorithme 4.1: Algorithme pour calculer l'histogramme h_i

1. Initialisation de l'histogramme h
2. Calcul de l'histogramme à partir de l'échantillon *E*
   Pour tout pixel *p* de l'échantillon E
   faire
   $h(p[C_1,C_2]) = h(p[C_1,C_2]) + 1$
   fin faire

La figure 4.8 illustre le principe de construction d'un histogramme de couleur de peau en utilisant les masques binaires. Par souci de clarté, nous avons utilisé ici une seule composante des espaces de couleur pour illustrer le principe.

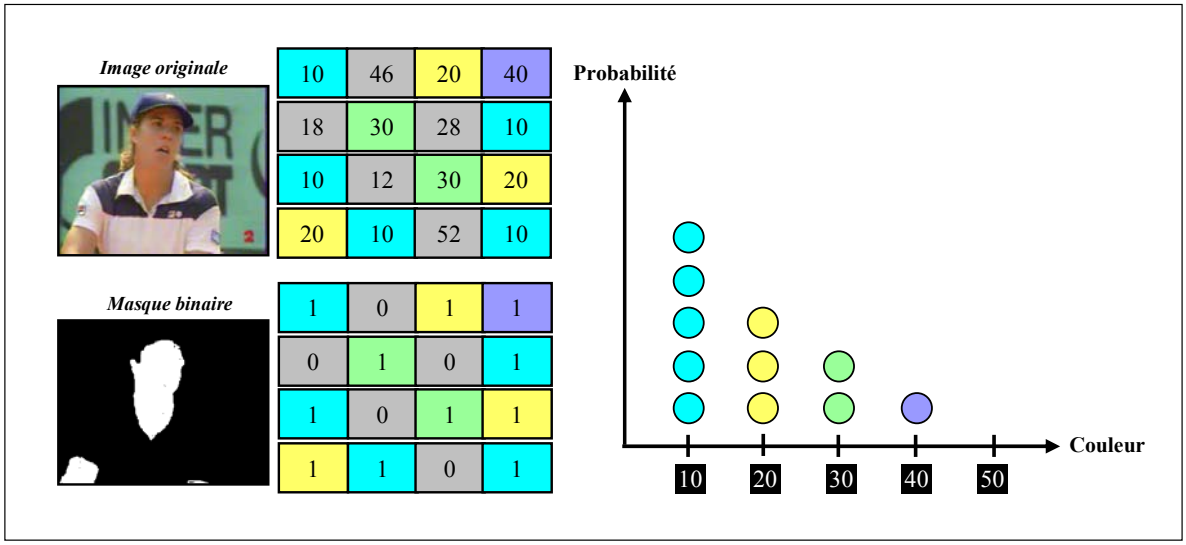


Figure 4. 8. *Principe de construction d'histogramme de couleur de peau à partir des masques*

Dans la figure 4.9 nous présentons quelques uns des 78 histogrammes de couleur de peau calculés sur le corpus de Compaq.

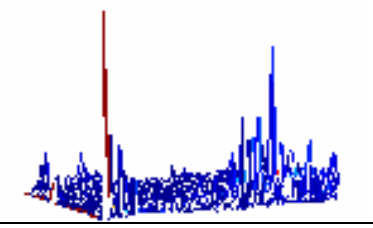
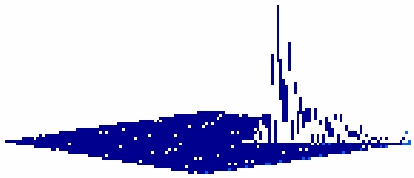
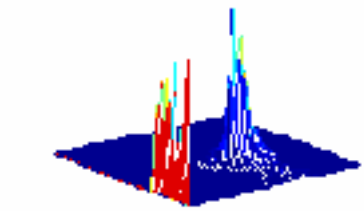
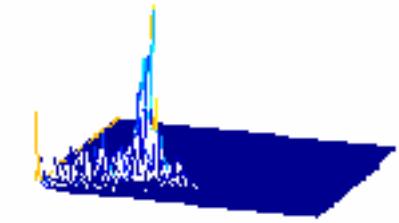
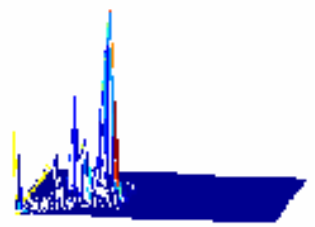
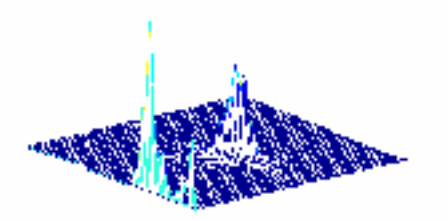
	
Histogramme N° 60	*Histogramme N° 2*
	
Histogramme N° 33	*Histogramme N° 14*
	
Histogramme N° 46	*Histogramme N° 11*

Figure 4. 9. *Histogrammes de couleurs selon différents axes de couleurs*

Après le calcul des histogrammes de couleur de pixels de peau et de non-peau, nous calculons la probabilité conditionnelle pour chaque composante sachant que cette couleur est une couleur de peau ou non. Les valeurs contenues dans les histogrammes sont alors converties en distribution discrète *P(.)* :

$$P(C_1, C_2 / peau) = \frac{P[C_1, C_2]}{T_P} \quad (4.\ 7)$$

$$P(C_1, C_2 / non.peau) = \frac{N[C_1, C_2]}{T_N} \quad (4.\ 8)$$

$P[C_1, C_2]$ est le nombre de pixels associé à une case (bins) de l'histogramme de peau formé par les deux axes C_1 et C_2. $N[C_1,C_2]$ est calculé de la même manière pour l'histogramme non-peau. T_P et T_N représentent respectivement le nombre total de pixels dans l'histogramme de peau et de non-peau.

Ceci nous permettra de calculer la probabilité qu'un pixel d'une composante de couleur donnée appartienne à la classe de peau. La probabilité P ($peau/C_1C_2$) est donnée par la formule de Bayes suivante:

$$p(peau / C1C2) = \frac{P(C1C2 / peau).P(peau)}{P(C1C2 / peau).P(peau) + P(C1C2 / non.peau).P(1 - P(peau))} \quad (4.\ 9)$$

Avec

$$0 \le P(peau / C_1C_2) \le 1$$

$$P(peau) + P(non.peau) = 1$$

$$P(peau) = \frac{T_P}{T_P + T_N}$$

Nous calculons ainsi la probabilité pour chaque pixel d'une image qu'il soit peau ou non peau, et cela suivant chaque combinaison d'axes. Notre but est à la fois d'identifier un pixel de peau avec un degré élevé de précision, et de déterminer les meilleurs axes.

Toutefois, si lors de la prise de décision on se base uniquement sur les probabilités, on peut être confronté à un pixel *x* ayant une probabilité élevée suivant une combinaison d'axes et une probabilité faible suivant d'autres combinaisons. Dans ce cas nous sommes en présence d'une ambiguïté que nous avons essayée de résoudre en utilisant les outils de data mining que nous présenterons dans la section suivante. L'objectif est de construire un graphe d'induction qui peut se réécrire sous forme de règle de production du type :

Si P (x/ C_1C_2) < c1 OU (P (x/ C_1C_2) > c1 et P (x/ C_3C_4) < c2)
ALORS classe Non peau
Si P (x/ C_1C_2) > c1 ET P (x/ C_3C_4) > c2
ALORS classe peau

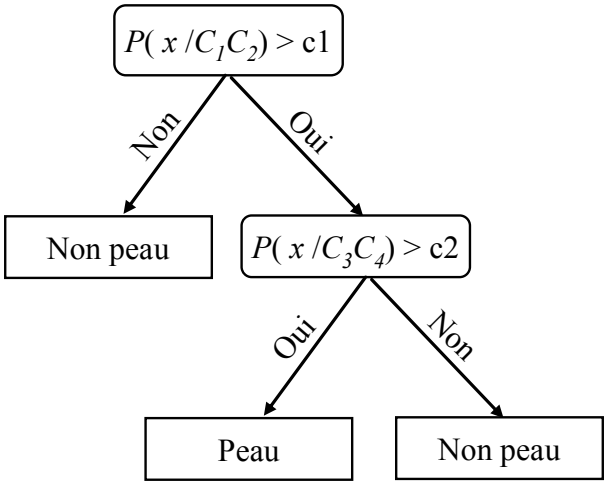


Figure 4. 10. *Type de Graphe d'induction pour l'approche bayesienne*

Mais cette approche a vite trouvé ses limites compte tenu de l'importance de la combinaison d'axes. En plus, l'apprentissage sur des données issues de probabilités sur une combinaison d'axes ne permet pas de discriminer le poids de chacun, et ne permet pas une efficacité lors de l'apprentissage.

Au final, la qualité du modèle de prédiction obtenu par cette approche n'est pas satisfaisante, d'une part les règles obtenues ne nous permettent pas d'améliorer la performance de la méthode développée par Compaq et d'autre part elles sont complexes.

Cela nous a amené à exploiter directement les valeurs des pixels issues des différents axes de représentation afin de déterminer la pertinence de chaque axe indépendamment des autres, et d'en extraire les règles de prédiction.

4.5.1.2 *Exploitation directe des valeurs de pixels*

Pour déterminer l'espace de couleur le mieux adapté, nous définissons pour chaque classe C_j, j désignant respectivement la classe de peau (p) ou de non peau (np), un ensemble de N_w pixels représentatifs $w_{i,j}$, i= 1,…, N_w. Ces pixels représentatifs sont extraits à partir des images d'apprentissage et de leurs masques binaires.

Dans un espace de couleur de dimension d= 17, nous caractérisons chaque pixel $w_{i,j}$, représentatif de la classe C_j, par une observation $X_{i,j} = \left[x_{i,j}^1, ..., x_{i,j}^k, ..., x_{i,j}^d\right]^T$ où $x_{i,j}^k$ est le niveau du k^{eme} axe (composante de couleur).

Sachant que les méthodes de traitement de données exigent souvent une présentation particulière des fichiers de données, nous représentons l'ensemble des observations comme décrit dans le tableau 4.2.

Tableau 4. 2. *Extrait de fichier d'apprentissage*

Variables exogènes					Classe
$x_{1,p}^{1}$	…	$x_{1,p}^{k}$	…	$x_{1,p}^{d}$	1
⋮	…	⋮	…	⋮	⋮
$x_{N_w,p}^{1}$	…	$x_{N_w,p}^{k}$	…	$x_{N_w,p}^{d}$	1
⋮	…	⋮	…	⋮	⋮
$x_{1,np}^{1}$	…	$x_{1,np}^{k}$	…	$x_{1,np}^{d}$	0
⋮	…	⋮	…	⋮	⋮
$x_{N_w,np}^{1}$	…	$x_{N_w,np}^{k}$	…	$x_{N_w,np}^{d}$	0

Les lignes du tableau correspondent aux pixels représentatifs, tandis que les colonnes correspondent aux axes de représentation testés. Les différentes composantes couleur utilisées sont décrites par le tableau 4.3.

Tableau 4. 3. *Variables exogènes utilisées*

R	Red	Rouge
G	Green	Vert
B	Blue	Bleu
r	Normalized R	Rouge normalisé
g	Normalized G	Verte normalisé
b	Normalized B	Bleue normalisé
H	Hue	Teinte
S	Saturation	Saturation
V	Value	Luminance
Y	Illumination	Luminance
I	Inphase	Interpolation
Q	Quadrature	Quadrature
Cr	Chrominance	Chrominance
Cb	Chrominance	Chrominance
C	Cyan	Cyan
M	Magenta	Magenta
Y	Yellow	Jaune

A cet ensemble de données nous avons ajouté une autre variable que nous trouvons pertinente. Il s'agit de la distribution spectrale que nous présentons dans la section suivante. Un extrait de notre fichier de données est présenté par la figure 4.11.

```
P     V1          V2          V3          V4          V5          V6          C
1,0.061055,0.031818,0.155225,0.071557,0.056528,0.071690,..., 1
2,0.071105,0.172834,0.192698,0.082228,0.167058,0.082224,..., 1
3,0.049195,0.022229,0.048231,0.054901,0.028214,0.055375,..., 1
4,0.049195,0.022229,0.048231,0.054901,0.028214,0.055375,..., 1
5,0.071105,0.172834,0.192698,0.082228,0.167058,0.082224,..., 0
6,0.054431,0.022827,0.046170,0.061891,0.027777,0.062345,..., 0
7,0.049195,0.020797,0.024770,0.054901,0.028214,0.055375,..., 1
8,0.049195,0.028440,0.038839,0.054901,0.028214,0.055375,..., 1
9,0.071105,0.172834,0.192698,0.082228,0.167058,0.082224,..., 0
10,0.129661,0.421915,0.329804,0.339445,0.383743,0.30221,..., 1
```

Figure 4. 11. *Extrait du fichier de données préparé pour l'apprentissage*

L'objectif est donc de construire un graphe d'induction qui peut se réécrire sous forme de règle de production du type :

Si $V_1(x) < c1$ OU ($V_1(x) > c1$ et $V_2(x) < c2$)
ALORS classe Non peau
Si ($V_1(x) > c1$ et $V_2(x) > c2$)
ALORS classe peau

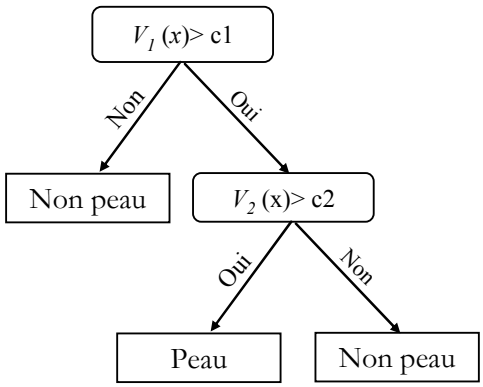


Figure 4. 12. *Type de graphe d'induction pour la 2ème approche*

4.5.2 Distribution spectrale

Les objets et les matières sont perçus par l'œil humain en fonction de la manière dont ils modifient la lumière qui les éclaire, alors que les sources de lumière restent visibles en raison de la lumière qu'elles émettent. Les objets ou les matériaux peuvent être de natures diverses comme par exemple, une surface peinte, une feuille de papier, un objet en matière plastique ou n'importe quel autre produit. Dans la mesure où la peau est une surface différente d'autres matières, nous avons eu l'idée d'étudier et dégager les caractéristiques de la peau d'un point de vue de la distribution spectrale.

On voit que la matière a seulement la propriété de réfléchir certains rayonnements électromagnétiques auxquels l'œil humain est sensible. La lumière qui éclaire la peau sera modifiée par son interaction avec cette dernière de multiples façons et dans des directions

diverses. La réflexion de cette distribution de lumière nous donne l'impression visuelle (apparence) de la peau (cf. figure 4.13).

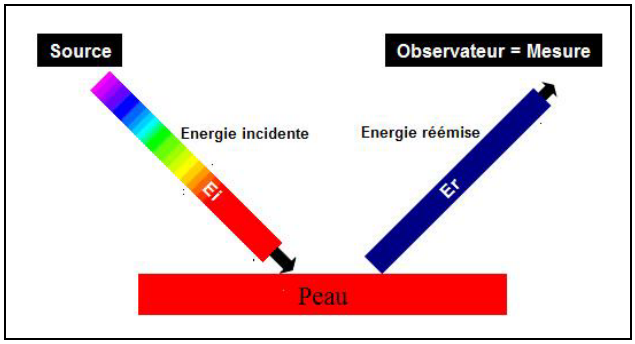


Figure 4. 13. *Interaction lumière -peau*

La distribution spectrale constitue un indice important pour l'identification des pixels. En effet, un pixel d'un objet est associé à la source qui l'éclaire. Les différentes sources lumineuses peuvent être estimées par une analyse de la distribution spectrale M_f du rayonnement et par une classification de la couleur qui lui est associée. Ainsi, une distribution spectrale M_f est présentée via 6 bandes spectrales visibles. La figure 4.14 présente ces différentes bandes spectrales. La quantification de la distribution spectrale d'un pixel est calculée à l'aide de la formule suivante :

$$M_f = M_R r_0 + M_V v_0 + M_B b_0 \qquad (4.10)$$

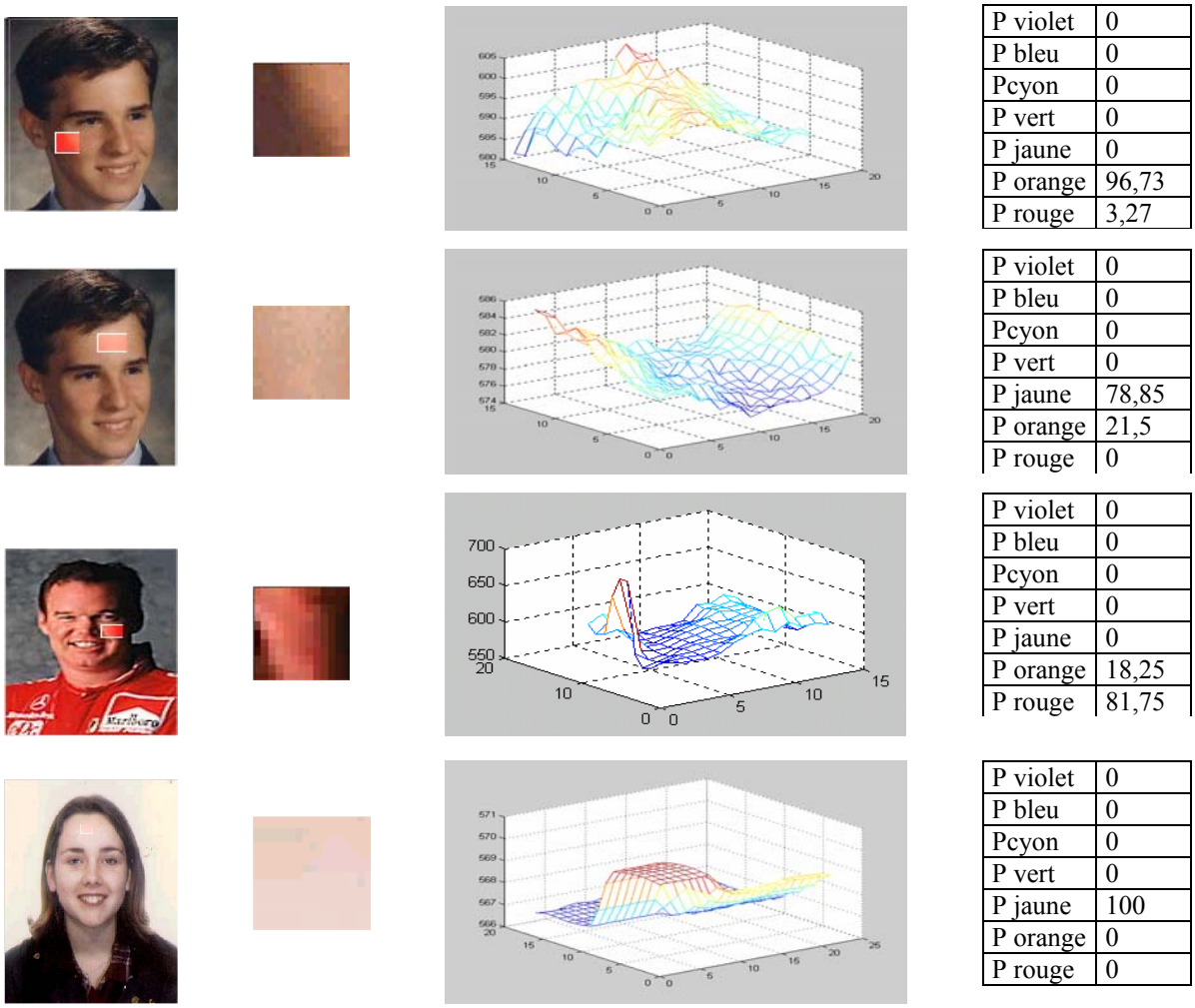
où M_R, M_v, M_b sont respectivement, les composantes rouge (R), vert (G) et bleu (B) d'un pixel, alors que $r_0 = 700$, $v_0 = 546.1$, $b_0 = 435.8$ donnent le spectre primaire R, G, B dans le système C.I.E.

Couleur	*Longueur d'onde*
Violet	380-450nm
Bleu	450-480nm
Cyan	480-490nm
Vert	490-560nm
Jaune	560-580nm
Orange	580-600nm
Rouge	600-700nm

Figure 4. 14. *Les longueurs d'ondes des six spectres visibles*

Cette distribution spectrale est d'une importance capitale lors de la classification des pixels de peau. Les expérimentations que nous avons conduites, montrent que la distribution spectrale calculée à partir d'un pixel de peau est généralement caractérisée par une bande spectrale de longueur d'onde comprise entre 568nm et 680nm. Cet intervalle caractérise en réalité trois bandes spectrales qui sont l'orange, le jaune et le rouge (cf. figure 4.15).

Afin d'étudier la pertinence de cette distribution, nous avons appliqué une classification supervisée sur un corpus de 463095 pixels. Le graphe de la figure 4.16 montre un taux de réussite de l'ordre de 79%.



P violet	0
P bleu	0
Pcyon	0
P vert	0
P jaune	0
P orange	96,73
P rouge	3,27

P violet	0
P bleu	0
Pcyon	0
P vert	0
P jaune	78,85
P orange	21,5
P rouge	0

P violet	0
P bleu	0
Pcyon	0
P vert	0
P jaune	0
P orange	18,25
P rouge	81,75

P violet	0
P bleu	0
Pcyon	0
P vert	0
P jaune	100
P orange	0
P rouge	0

Figure 4. 15. *Distribution spectrale pour les pixels de peau*

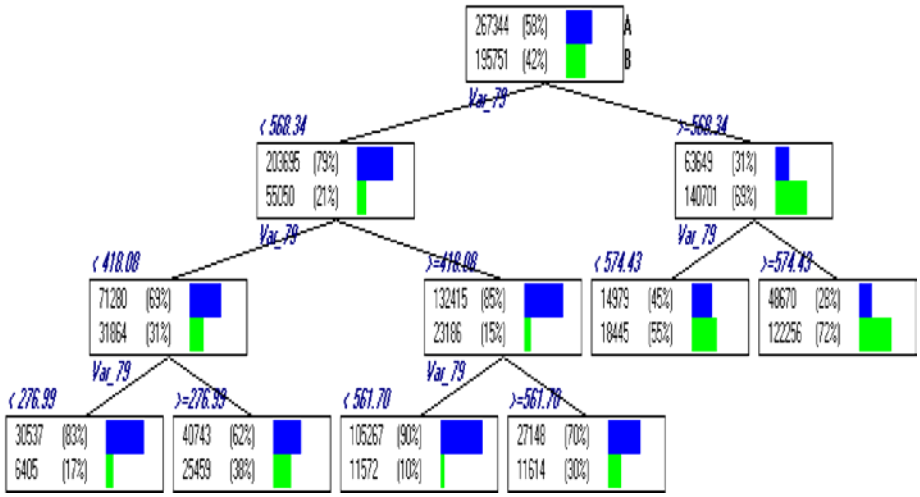


Figure 4. 16. *Graphe d'induction utilisant comme attribut la distribution spectrale (A : pixels de non peau et B : pixels de peau)*

4.5.3 Apprentissage supervisé pour l'extraction des règles de prédiction

Lors de la phase de préparation de données, on associe à chaque pixel w de notre base d'apprentissage, sa classe $C(w)$ qui peut être classe peau ou classe non-peau. On dit que la variable C prend ses valeurs dans l'ensemble des classes $\not\subset$={peau, non-peau}.

$$C : \Omega \rightarrow \not\subset = \{\text{peau, non-peau}\}$$
$$w \rightarrow C(w)$$

Comme on a vu précédemment, la détermination de C(w) n'est pas facile pour des raisons diverses telles que les conditions d'éclairage, les différentes races etc. C'est pour cette raison que nous cherchons un modèle de prédiction φ permettant d'identifier la classe C d'un pixel dont on ne connaît que ces variables exogènes calculées dans la phase de préparation de données. Avant de procéder à l'apprentissage par des algorithmes de data mining, une sélection préalable d'un sous-ensemble de variables optimal est nécessaire pour la mise en œuvre des algorithmes d'apprentissage.

4.5.3.1 *Sélection des variables*

Dans cette phase, nous cherchons à déterminer les variables qui ont une influence sur notre problème. La sélection des variables contribue à réduire la taille du problème en isolant les variables exogènes les plus pertinentes. L'élimination des variables inutiles et redondantes permet d'accélérer le processus d'apprentissage et d'augmenter la fiabilité du classifieur obtenu. L'observation des corrélations entre certaines données peut également aboutir à une réduction du nombre des variables.

Cette réduction de la complexité initiale permet d'optimiser notre modèle de peau. Toutefois, elle pose le problème du choix des variables pertinentes et aptes à modéliser la peau.

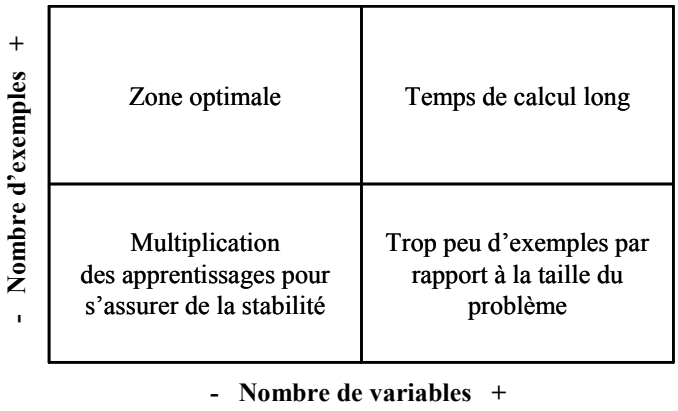


Figure 4. 17. ***Liaison entre dimension et exemples***

Afin de déterminer l'espace de couleur optimal pour l'identification de pixels de peau, nous avons utilisé une approche de type filtre (cf. figure 4.18) et plus précisément la méthode Relief [169]. Cette dernière fournit des renseignements sur la pertinence d'une variable par rapport aux autres. Elle utilise les caractéristiques générales de l'ensemble d'apprentissage pour classer les variables en attribuant un poids, à chacune d'entre elles, compris entre -1 et 1. L'algorithme 4.2 rappelle le principe de la méthode Relief.

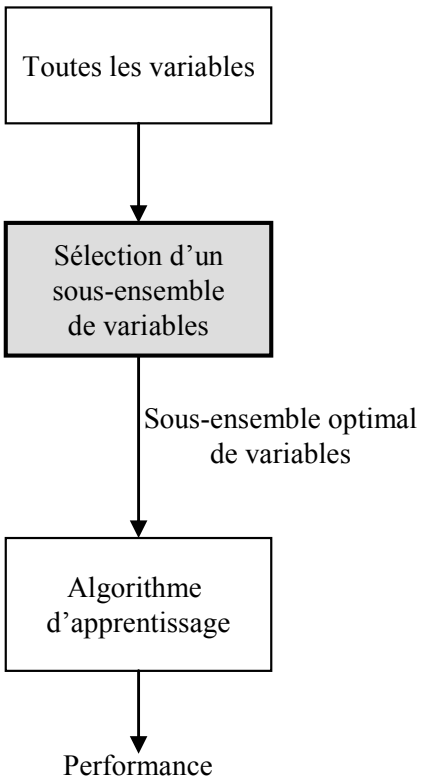


Figure 4. 18. *Approche Filtre*

Algorithme 4.2 : Algorithme de sélection de variable 'Relief'

Entrée X ensemble total des variables initiales
Ω ensemble des instances
Sortie X' ensemble de variables sélectionnées

Début Relief
Fixer un seuil τ pour filtrer les variables ayant un poids supérieur ou égal.
$X' = \phi$
Tirer aléatoirement un échantillon $\Omega' \subseteq \Omega$;
Initialiser tous les poids $w_j ; j = 1, \cdots, p$ à zéro
Pour $t = 1, T$ /*T est le nombre d'itérations choisi arbitrairement */
Choisir aléatoirement une instance $\omega \in \Omega'$
Chercher sa plus proche instance $\omega_=$ de la même classe et sa plus proche instance $\omega_{\neq}$ de classe différente :
Pour $j = 1, ..., p$

$$w_j = w_j - \frac{\delta\left(X_j(\omega), X_j(\omega_=)\right)}{T} + \frac{\delta\left(X_j(\omega), X_j(\omega_{\neq})\right)}{T}$$

Fin Pour
Fin Pour

Pour $j = 1, r$
Si $w_j \geq \tau$ alors $X' = X' \cup \{X_j\}$
Retourner X'.
Fin Relief.

Avec : $\delta\left(X_j(\omega), X_j(\omega')\right) = \dfrac{\left|X_j(\omega) - X_j(\omega')\right|}{\left|\max_{X_j} - \min_{X_j}\right|}$

Nous avons commencé par choisir un échantillon d'instances composé de 3 412 992 pixels du corpus CRL. Nous avons cherché ensuite pour chaque instance la plus proche instance de réussite et la plus proche instance d'échec en se basant sur une mesure de distance. L'instance de réussite la plus proche est l'instance qui est à la plus petite distance parmi toutes les instances appartenant à la même classe que l'instance choisie. L'instance d'échec la plus proche est l'instance qui a la plus petite distance parmi toutes les instances appartenant à une classe différente de celle de l'instance choisie.

Les poids des différentes variables, qui sont initialisés à zéro au début, sont mis à jour à chaque itération. Le nombre d'itérations est fixé à 100 000. Cette démarche est basée sur l'idée intuitive suivante : une variable est plus pertinente qu'une autre si elle distingue une instance de son instance d'échec la plus proche, et moins pertinente si elle distingue une instance de son instance de réussite la plus proche.

Les résultats de classement des différentes variables sont présentés dans le tableau 4.4.

Tableau 4. 4. *Résultats de l'algorithme Relief*

Variables exogènes	Poids
H	0,00184
Dist	0,00050
r	0,00049
S	0,00036
b	0,00035
Cb	0,00018
I	0,00018
g	-0,00002
Cr	-0,00011
Q	-0,00030
R	-0,00048
Y	-0,00117
G	-0,00119
V	-0,00122
C	-0,00134
M	-0,00139
B	-0,00155

A ce stade d'étude, il n'y a pas un espace de couleur classique qui émerge et se distingue clairement des autres. Toutefois nous remarquons que les composantes issues des deux espaces HSV et rgb (RGB normalisé) occupent les premières places, ainsi que la distribution spectrale (Dist), conformément à nos prédictions. Ce classement nous guidera pour le choix de la stratégie à adopter pour déterminer l'espace de couleur et les règles de décision.

Nous avons également calculé pour chaque espace de couleur utilisé les poids cumulés de leurs différentes composantes (tableau 4.5). Ces résultats confirment notre première remarque.

Tableau 4. 5. *Poids pour chaque espace de couleur classique*

Espaces de couleurs	Poids
HSV	$9{,}8\ 10^{-04}$
rgb	$8{,}2\ 10^{-04}$
Dist	$5\ 10^{-04}$
YCrCb	$-1{,}10\ 10^{-03}$
YIQ	$-1{,}29\ 10^{-03}$
RGB	$-3{,}22\ 10^{-03}$
CMY	$-3{,}9\ 10^{-03}$

4.5.3.2 *Extraction des règles de décision*

Afin de trouver le meilleur modèle de prédiction, nous avons testé plusieurs techniques basées sur les graphes d'induction ID3, C4.5, SIPINA avec λ=12 et une technique basée sur les réseaux de neurone. Pour cette dernière nous avons utilisé un perceptron à 2 couches cachées, composée chacune de 30 neurones, dont l'algorithme d'apprentissage est l'algorithme de propagation d'erreur classique. Les expérimentations ont été effectuées sur une base de test différente de celle qui a servi pour l'apprentissage, où nous avons calculé à

chaque fois le taux de vrais positifs (VP), le taux de faux positifs (FP), et le taux d'erreur globale. Cette base de test est composée de 19 112 758 pixels extraits du corpus CRL de Compaq.

Dans un premier temps nous avons voulu nous assurer de la pertinence des variables obtenues lors de la phase précédente. Nous avons donc effectué deux séries d'apprentissage, l'une en utilisant toutes les variables et l'autre uniquement avec les variables ayant un poids positif (c.à.d. S, Dist, r, H, b, Cb, I). Les tableaux 4.6 et 4.7 récapitulent les résultats obtenus sur chacune des deux séries et selon les différents algorithmes.

Tableau 4. 6. *Résultats de l'utilisation de toutes les variables*

Variables testées	Toutes les variables			
Algorithmes	C4.5	ID3	SIPINA	RN
Taux d'erreur Globale (TE)	18,77	18,39	17,78	21,17
Taux de vrais positifs	89,13	89,45	89,52	78,37
Taux de faux positifs	19 %	18,62	17,98	21,16

Tableau 4. 7. *Résultats de l'utilisation des variables ayant un poids positifs*

Variables testées	H, Dist, r, S, b, Cb, I			
Algorithmes	C4.5	ID3	SIPINA	RN
TE Globale	18,51	18,45	17,33	20,47
Taux de vrais positifs	89,66	89,45	89,65	80,84
Taux de faux positifs	18,74	18,68	17,54	20,51

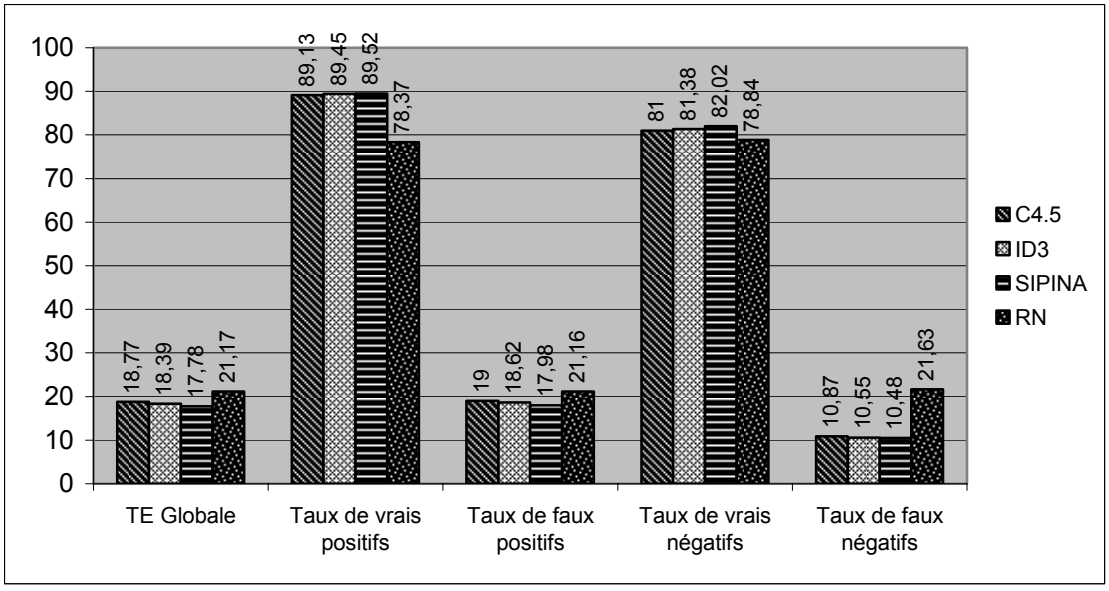


Figure 4. 19. *Résultats obtenus à partir de toutes les variables selon différents algorithmes*

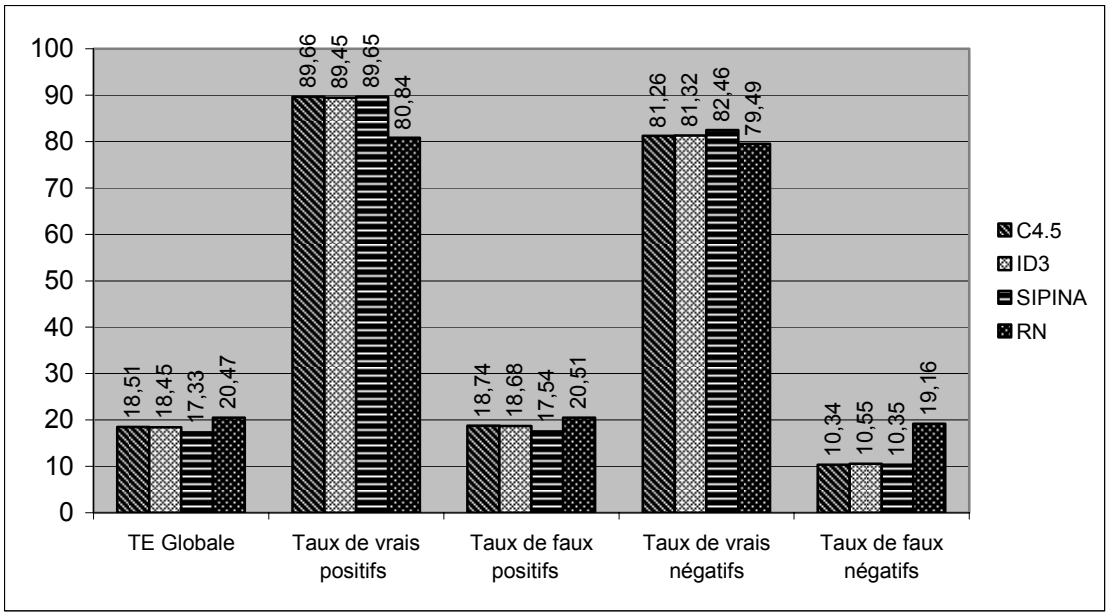


Figure 4. 20. *Résultats obtenus à partir des variables ayant un poids positifs*

On peut remarquer que la réduction du nombre de variables aux variables ayant un poids positif a permis de diminuer un peu le taux d'erreur.

Par ailleurs, nous nous sommes intéressés à l'étude des différentes combinaisons possibles entre les trois espaces de représentation les mieux classées, à savoir HSV, rgb et Dist. Le tableau 4.8 présente les résultats obtenus.

Tableau 4. 8. *Résultats de différentes combinaisons entre rgb, HSV et Dist, selon différents algorithmes*

C4.5

Variables testées	HSV	rgb	HSV+Dist	rgb+Dist	HSV+rgb+ Dist
TE Globale	18,25	21,16	15,66	19,58	17,21
Taux de vrais positifs	89,54	89,92	89,45	89,46	89,81
Taux de faux positifs	18,47	21,48	15,81	19,84	17,42

ID3

Variables testées	HSV	rgb	HSV+Dist	rgb+Dist	HSV+rgb+ Dist
TE Globale	18,51	19,92	18,27	19,29	18,46
Taux de vrais positifs	89,55	88,58	89,44	88,71	89,65
Taux de faux positifs	18,74	20,16	18,49	19,51	18,69

Sipina

Variables testées	HSV	rgb	HSV+Dist	rgb+Dist	HSV+rgb+ Dist
TE Globale	16,59	19,51	13,78	15,95	14,31
Taux de vrais positifs	89,62	88,75	89,71	89,21	89,51
Taux de faux positifs	16,77	19,75	13,88	16,11	14,42

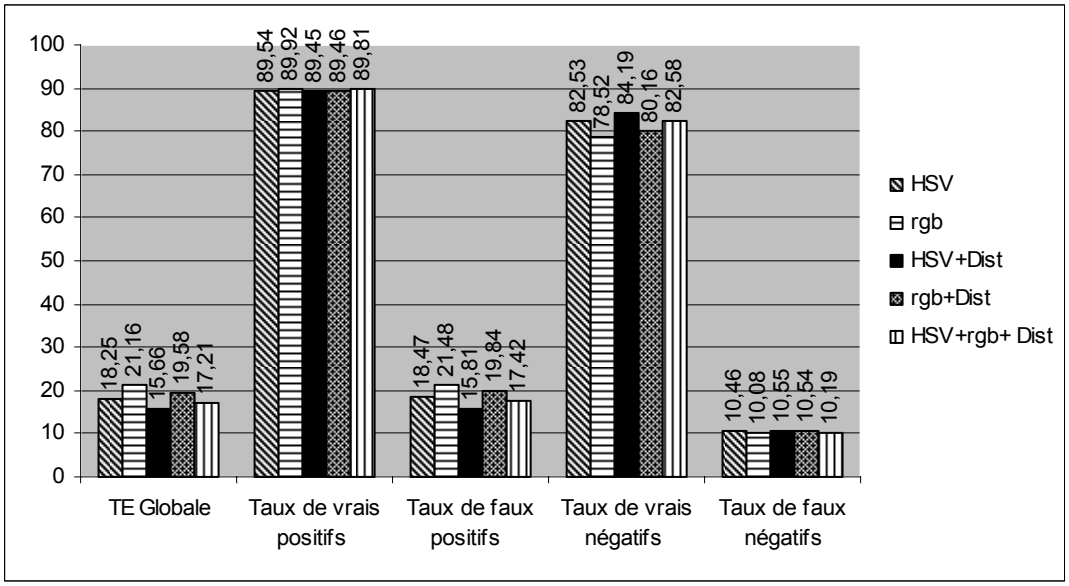


Figure 4. 21. *Résultats obtenus par l'algorithme C4.5*

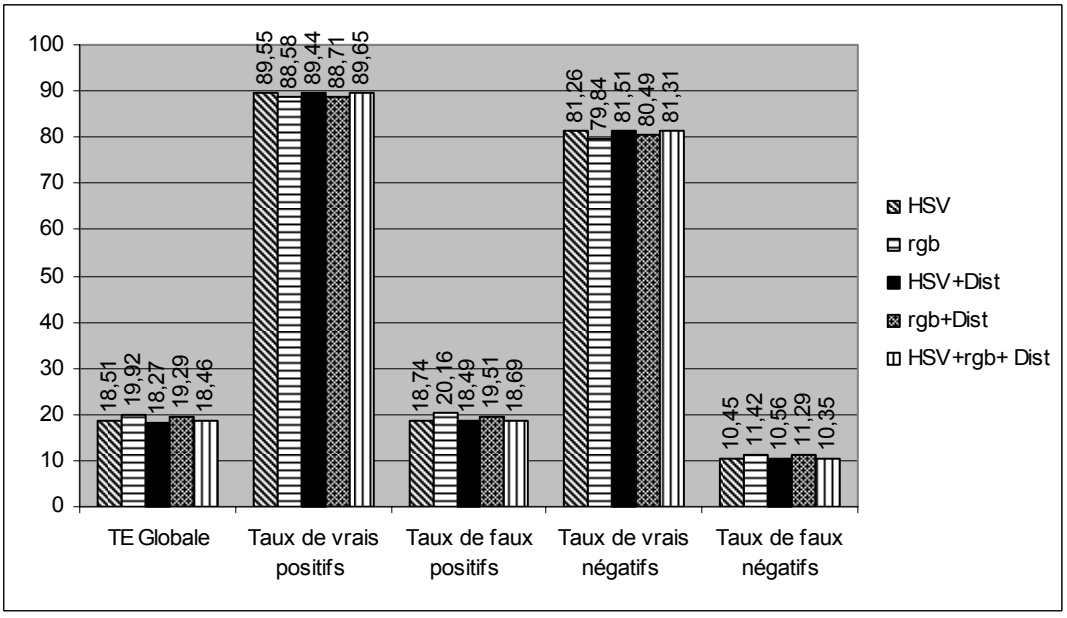


Figure 4. 22. *Résultats obtenus par l'algorithme ID3*

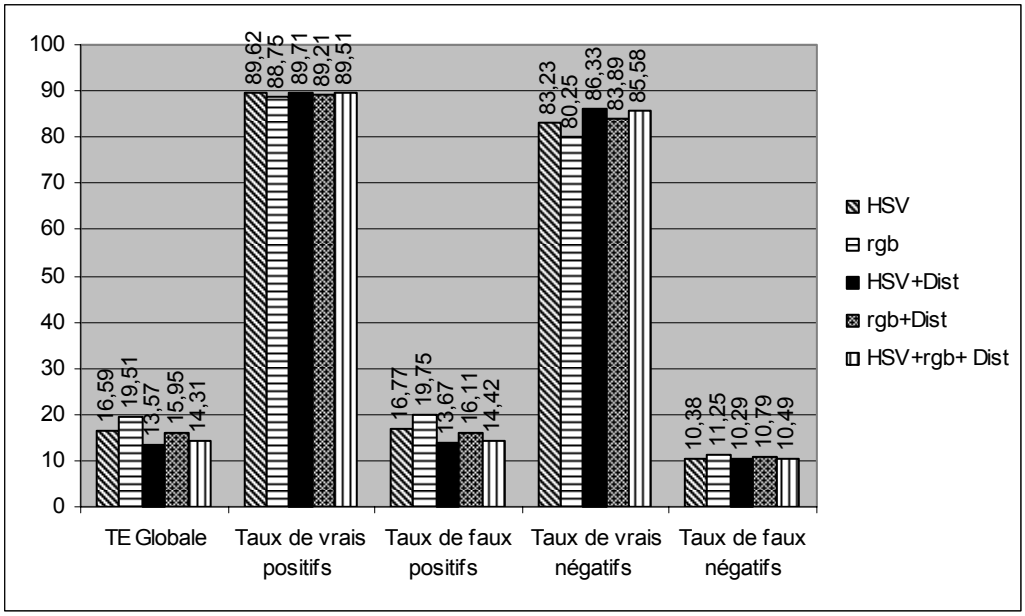


Figure 4. 23. *Résultats obtenus par l'algorithme SIPINA*

A partir de cette série de tests, il apparaît que l'espace hybride composé des composantes H,S,V et Dist produit les meilleurs résultats, notamment avec l'algorithme SIPINA. L'algorithme suivant décrit son processus de construction du graphe d'induction.

Algorithme 4.3 : Algorithme de construction du graphe d'induction

1. Choix de la mesure d'incertitude I. Nous pouvons utiliser la formule la plus simple, celle qui utilise l'entropie quadratique :

$$I_\lambda(S_i) = \sum_{j=1}^{K} \frac{n_j}{n} \left(- \sum_{i=1}^{m} \frac{n_{ij} + \lambda}{n_j + m\lambda} \left(1 - \frac{n_{ij} + \lambda}{n_j + m\lambda} \right) \right)$$

2. Nous déterminons la valeur des effectifs minimaux τ exigés à chaque sommet du graphe

3. Nous déterminons la valeur de λ suivant la procédure décrite au chapitre 3

4. On considère la partition grossière S_0, pour laquelle nous déterminons T_0 le tableau de contingence T_0 et nous calculons la mesure d'entropie $I_\lambda(S_0)$.

5. On cherche parmi les p variables exogènes $X_1 \ldots, X_p$, celle qui engendre la meilleure partition. Si on note S_1^j la partition engendrée par X_j et T_1^j le tableau de contingence associé, nous aurons à déterminer S_1 telle que $\left(I_\lambda(T_0) - I_\lambda(T_1)\right) = \max_{j=1}^{p} \left(I_\lambda(T_0) - I_\lambda(T_1^j)\right)$

6. soit i l'indice de l'itération, i=1.

7. l'itération courante consiste à appliquer la procédure de passage de la partition S_i à S_{i+1}. On cherche la partition S_{i+1} telle que $I_\lambda(S_{i+1}) < I_\lambda(S_i)$.

4.6 Segmentation de l'image en régions de peau

Le modèle de peau résultant du processus précédent conduit tout de même à des erreurs de classification quand il est appliqué directement à une image de couleur (cf. figure 4.25). Par ailleurs, de nombreuses applications utilisant un modèle de peau ont besoin d'une connaissance de régions de peau. Aussi, nous procédons à une étape de segmentation dont l'objectif de la procédure de segmentation consiste à extraire des régions significatives représentant des objets d'intérêt composés de pixels de peau.

Pour cela, on a besoin d'une bonne méthode de segmentation qui est capable de déterminer les régions cohérentes à la fois visuellement et du point de vue de leur contenu pour faciliter leur bonne interprétation. Il existe deux familles principales de méthodes de segmentation : celles qui sont basées « contour » qui tentent de rechercher des pixels contours couleur correspondant aux variations locales significatives des couleurs des pixels, et celles qui sont basées « régions » qui recherchent dans l'image des sous ensembles de pixels connexes dont les couleurs sont homogènes.

Nous proposons d'exploiter les règles obtenues à partir du graphe d'induction pour extraire des régions homogènes et cohérentes de peau. Dans les sections suivantes, nous présentons deux techniques de segmentation d'images, l'une basée sur la croissance de régions et l'autre sur la ligne de partage des eaux (LPE).

4.6.1 Croissance de régions

Le principe consiste à assembler tous les pixels voisins similaires selon le voisinage et selon la similarité visuelle déterminée à partir des règles de prédiction. En effet, le processus commence avec un pixel de départ et essaie d'attirer tous les pixels voisins à cette région qui s'agrandit au fur et à mesure.

L'algorithme choisit le meilleur pixel candidat parmi tous les pixels situés dans le voisinage de la région. La région continue à croître jusqu'à ce qu'il n'y ait plus de pixels candidats. Le parcours des pixels se fait dans l'ordre "escargot" qui favorise la compacité de la région (cf. figure 4.24). En effet, ce parcours a pour effet de remplir d'abord l'entourage du pixel candidat dans la région durant sa construction.

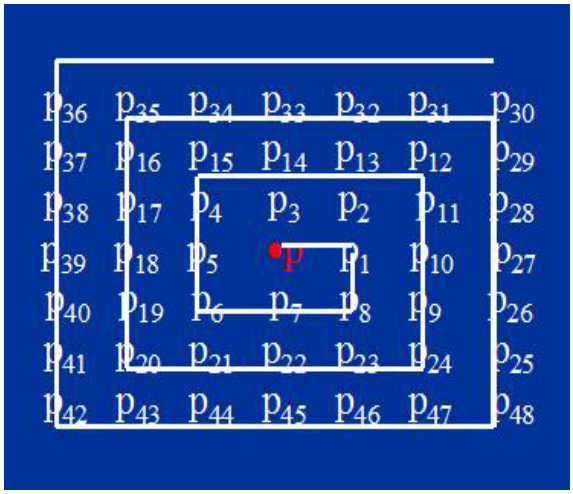


Figure 4. 24. *Parcours* "escargot" *pour la segmentation des régions de peau*

L'homogénéité d'une région R_i est définie par un prédicat d'uniformité et de cohérence, noté Pred (R_i). Ce prédicat est vrai si R_i est homogène et cohérente, faux dans le cas contraire. Pred(R_i) est Vrai si tous les pixels de la région R_i sont de pixels de peau et la taille de R_i est au delà d'un certain seuil. En fait, une région homogène n'est considérée comme cohérente qu'à la condition de représenter plus de λ % de l'image.

L'algorithme 4.4 décrit les différentes étapes de notre technique de segmentation de l'image en régions homogènes de peau.

Algorithme 4.4 : Algorithme de croissance de régions

Entrée :
P : germe potentiel dans l'image I;
ρ: seuil de voisinage (de balayage) en nombre de pixels. Par défaut, il est fixé à 1.
Pred : Prédicat composé des règles obtenus dans la phase de data mining, et de λ

Sortie :
La région cohérente R contenant le pixel de départ P, sinon NULL;

Initialisation :
Empiler le pixel P dans VCN ;
xi = xs = P.x; yi = ys = P.y; {Coordonnées du rectangle minimum englobant RME}
sx_centroid = 0; sy_centroid = 0; {Coordonnées du centre de gravité}
nb_pixels = 0; {Nombre total de pixels }
Couleur_P= couleur d'affichage des pixels de peau; {Couleur de la région pour l'affichage}

Etape 1: {Un nouveau pixel de couleur de peau est ajouté à la région}
Dépiler un pixel Q de la pile VCN;
Ajouter le pixel Q dans l'ensemble R;
Marquer le pixel Q comme intégré;
Mise à jour des coordonnées du RME
sx_centroid += Q.x; sy_centroid += Q.y; {Mise à jour du barycentre}

Etape 2 : {attraction des pixels voisins qui sont candidats}
Pour chaque pixel T voisin de Q, situé dans le périmètre ρ
Si T n'est pas encore intégré et T vérifie les règles de décision μ alors empiler T dans VCN ;

Etape 3 : {boucle}
Répéter les étapes 1 et 2 jusqu'à ce que la pile VCN soit vide;

```
Etape 4 : {Objet homogène O extrait}
    Si (nb_pixels /Taille image) > λ Alors
        O.region = R;
        O.couleur = couleur_P;{Couleur représentative de la couleur}
        O.surface = nb_pixels/Taille image;{calcul de la surface de la région}
        O.x_centroid = sx_centroid /nb_pixels; O.y_centroid = sy_centroid /nb_pixels;{calcul du
barycentre}
    Fin
```

La figure 4.25 montre l'apport du processus de segmentation d'image dans l'identification des régions de peau. Une telle segmentation va réduire quelques fausses détections qui ont été produites par notre modèle.

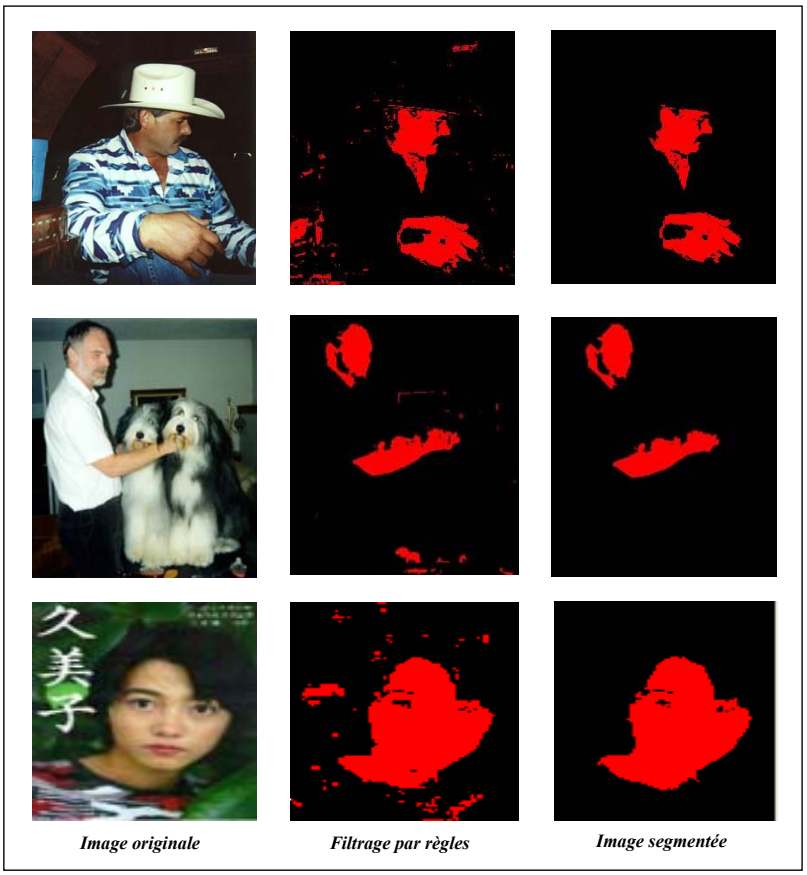


Figure 4. 25. *Apport de la segmentation*

4.6.2 La ligne de partage des eaux

La morphologie mathématique est à l'origine de la définition de la ligne de partage des eaux (LPE) (Watershed). Il y a plusieurs manières d'implémenter le principe de la LPE. Il existe deux classes principales d'implémentation : l'une est basée sur l'utilisation de fonctions de distances géodésiques [170] et une autre basée sur un algorithme récursif d'immersion [171] (algorithme de Vincent et Soille qui est intégrée à Matlab).

La mise en place de la lpe nécessite la mise en évidence des contours des objets. Cette détermination des extremas (minima ou maxima) utilise les variations du contraste de l'image, quantifiées par le calcul du gradient.

Dans le cadre de la ligne de partage des eaux on utilisera par défaut la 4-connexité, puisque la connexité 8 permet une multiplication d'extrema locaux (lors d'opérations morphologiques telles que l'érosion géodésique) et donc une sur-segmentation qui peut produire beaucoup plus de régions que celles marquées. Pour pallier cette sur-segmentation, on utilise en général la LPE avec les deux phases suivantes :

- Pré-Filtrage : Cela consiste à filtrer l'image originale afin de supprimer tous les minima non-significatifs de l'image gradient. En ce sens on cherche donc d'une certaine manière à moyenner localement l'image.

  Il existe plusieurs opérateurs de filtrage linéaires et non linéaires qui permettent un lissage de l'image tels que le filtre gaussien.
- Utilisation de marqueurs : elle consiste à choisir le nombre de minima locaux et donc les zones que l'on souhaite mettre en évidence grâce à la LPE et éliminer les informations non pertinentes.

  Dans cette partie, on suppose que l'on connaît un ensemble connexe de points faisant partie de l'objet d'intérêt ainsi qu'un ensemble de points de l'extérieur. Ces ensembles de points connexes sont appelés des marqueurs.
  On va alors modifier l'image en lui imposant que ces ensembles soient les uniques minima régionaux, chaque bassin devenant ainsi soit un unique objet, soit le fond de l'image.

  On cherche des ensembles connexes de points faisant partie des objets à segmenter. Quand les minima sont remplacés par des marqueurs il est primordial de contrôler la place de ces marqueurs. On doit choisir les marqueurs en s'assurant que les marqueurs contiennent les minima significatifs de l'image.

Les marqueurs de l'image d'origine ont été choisis à partir de transformations morphologiques de l'image de peau obtenues à partir des règles de décision et du reste de l'image (non peau avec fond de l'image). Ces opérations morphologiques ont pour but de pré-traiter l'image et éliminer les marqueurs (zones) non pertinents (non connexes et petits). Il existe plusieurs méthodes de type morphologique qui permettent de déterminer ces marqueurs. Une autre solution intéressante pour le choix des marqueurs est la délimitation grossière des zones d'intérêt de l'image source. Cette méthode s'applique lorsque la forme ou la position des objets à détourer est très complexe. Elle peut être manuelle ou le résultat d'une classification supervisée.

Dans notre contexte, on souhaite segmenter, localiser et extraire des informations uniquement des régions de peau tout en filtrant les petites régions et les régions non pertinentes. Pour cela, on a choisi la position des germes du gradient uniquement parmi des régions qui nous intéressent (zones de peau). En effet, on a choisi de délimiter les zones d'intérêt par gradient morphologique avec un élément structurant disk de taille donnée de l'image binaire (peau/non peau) prétraitée.

Nous avons utilisé un opérateur LPE basée sur l'inondation par file d'attente prioritaire (ensemble de FIFO ordonnées) [170] où il s'agit de traiter les pixels suivant leur proximité

spatiale tout en respectant l'ordre de leur niveau de gris (ou couleur). L'entrée de cet opérateur LPE est constituée de l'image à segmenter et l'image des marqueurs.

L'image des marqueurs est obtenue par calcul du gradient morphologique de l'image qui est une simple opération de différence entre le dilaté de l'image originale I par l'élément structurant symétrique B et l'érodé de I par B.

Le processus de segmentation d'une image est résumé dans la figure 4.26. En fait, cela correspond à un script d'enchaînement d'opérateurs de traitement d'image pour la localisation et la détection de régions homogènes de peau.

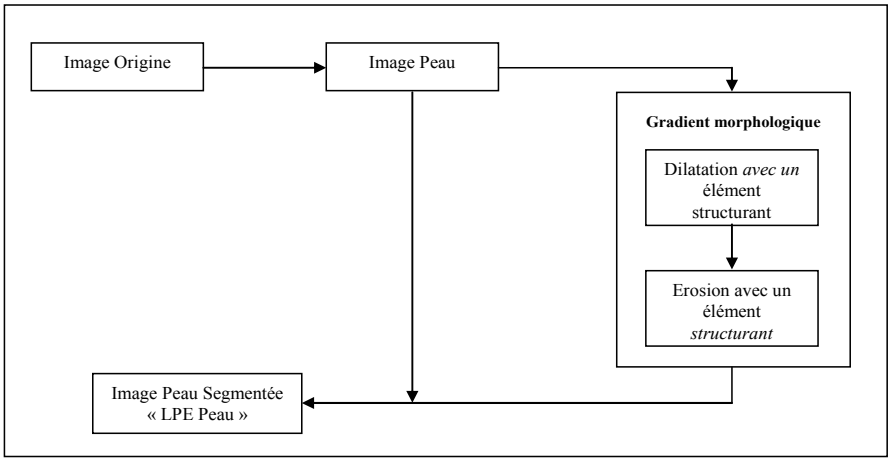


Figure 4. 26. *Segmentation de régions de peau homogènes par LPE*

La figure 4.27 montre un exemple qui présente ce processus de segmentation général à partir d'une image couleur.

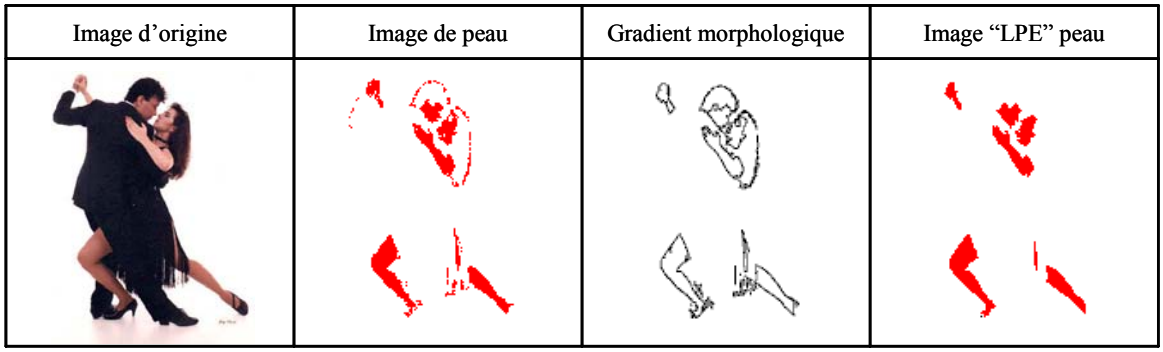


Figure 4. 27. *Différentes étapes de la segmentation par LPE*

Contrairement à la segmentation par l'approche contours, la segmentation par la Ligne de Partage des Eaux (LPE) conduit à des contours fins et fermés. C'est une approche intéressante qui pourra être exploitée ultérieurement en combinaison avec les règles de décision pour mettre en évidence des régions ou des formes bien précises. Néanmoins, cette méthode repose

sur le choix des marqueurs et peut fournir une sur-segmentation même si l'image originale est homogène. Ceci est dû au fait que les images gradient des images naturelles contiennent un grand nombre de minima (dû aux différentes variations locales des niveaux de gris ou de textures des régions). Chaque minimum génère un bassin versant dans la LPE.

On souhaite utiliser une méthode de segmentation rapide dans un processus de filtrage de sites, sur des images binaires de peau/non peau qui sont le résultat des règles de décision. En plus, on veut tenir compte d'autres critères tels que le voisinage lors du processus de segmentation. Pour toutes ces raisons, il nous a semblé plus judicieux d'utiliser la méthode de croissance de régions. En effet, cette dernière est rapide et convient mieux à une utilisation où il faut adapter les paramètres en fonction des applications puisque l'agglomération des pixels peuvent exploiter certaines connaissances a priori des images du fait que la décision d'intégrer à une région un pixel voisin repose sur des critères d'homogénéité imposés à la zone en croissance.

4.7 Expérimentations

Cette section présente les résultats expérimentaux de notre technique. La première étape de cette expérimentation consiste à montrer l'apport de notre modèle par rapport à celui de Compaq. Nous présentons les taux de vrais positifs (VP), les taux de faux positifs (FP), ainsi que les taux d'erreur globale pour notre méthode bayésienne (A), notre méthode retenue (B), et la méthode de Compaq (C). Ces expérimentations ont été effectuées sur notre base de test qui est indépendante de la base d'apprentissage et qui est précédemment décrite.

La figure 4.28 décrit les résultats obtenus.

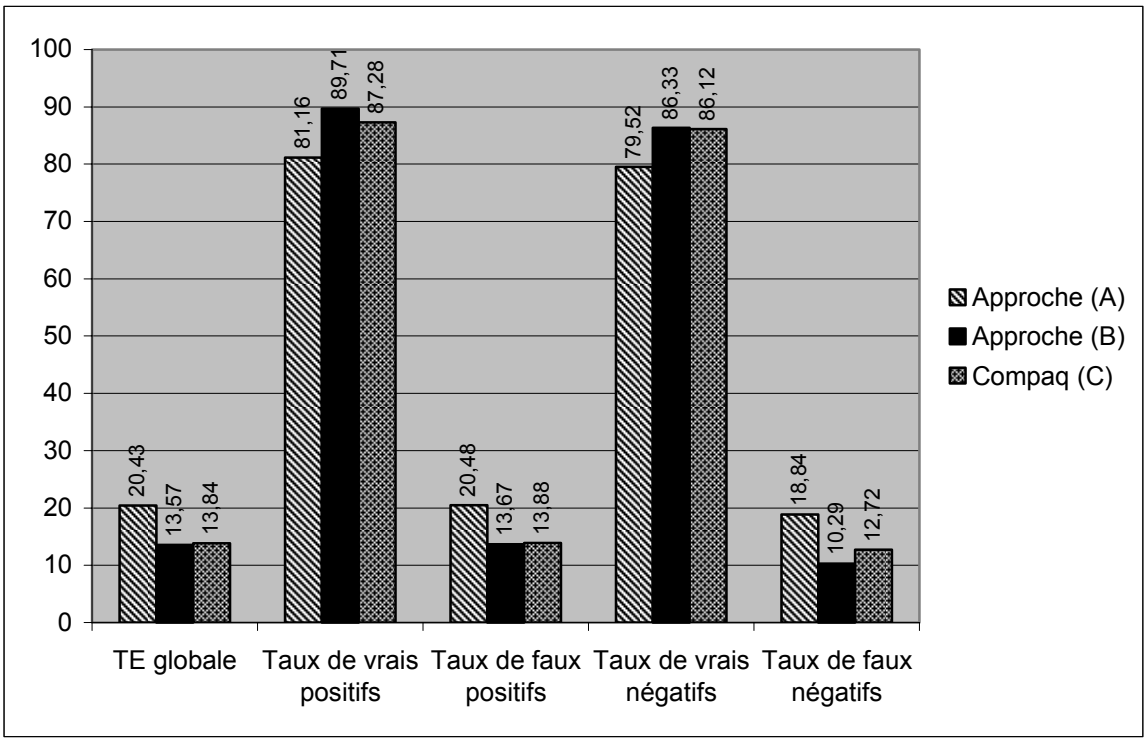
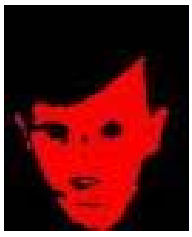
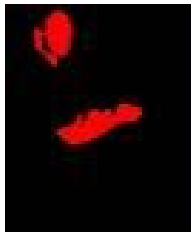
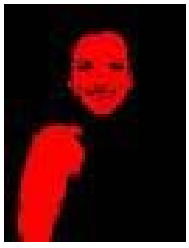
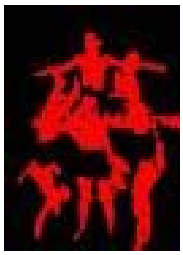
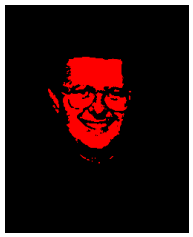
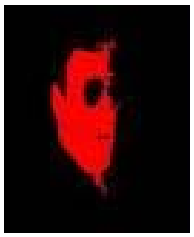


Figure 4. 28. *Comparaison résultats*

Comme illustré dans la figure 4.28, notre technique basée sur une approche bayésienne est moins performante que celle de Compaq [80]. En revanche, notre 2ème technique présente un taux de vrais positifs plus élevé de 2,43 %, et un taux de faux positifs légèrement inférieur (0,21%). Ces différences entre les taux présentés par les deux approches ne sont pas trop élevées mais montrent les améliorations apportées par notre nouvelle approche. Ces performances peuvent être assez significatives pour certaines applications. En effet, les pixels de peau identifiés comme non peau ne peuvent être récupérés une fois que le traitement est fait sur l'image. D'où l'intérêt de cette nouvelle approche qui minimise le taux de faux négatifs. Par ailleurs, le taux de faux positifs, c'est à dire les pixels non peau considérés comme peau, peut être minimisé par un traitement par région de l'image, comme cela a été décrit dans la section 4.6.

Nous présentons également les résultats obtenus après procédures de détection des pixels de peau et de segmentation en régions de l'image. Comme on peut le voir sur la figure 4.29, ces images sont très variées, comprenant des enfants, des adultes, des hommes, des femmes, des Asiatiques, des Européens, des Africains, des Latino-Américains. Ces résultats sont obtenus après fixation du seuil λ à 0.8, la taille minimum au delà de laquelle une région sera retenue (cf. figure 4.29).

Image cible	*Image résultat*	*Image cible*	*Image résultat*	*Image cible*	*Image résultat*
					
					
					

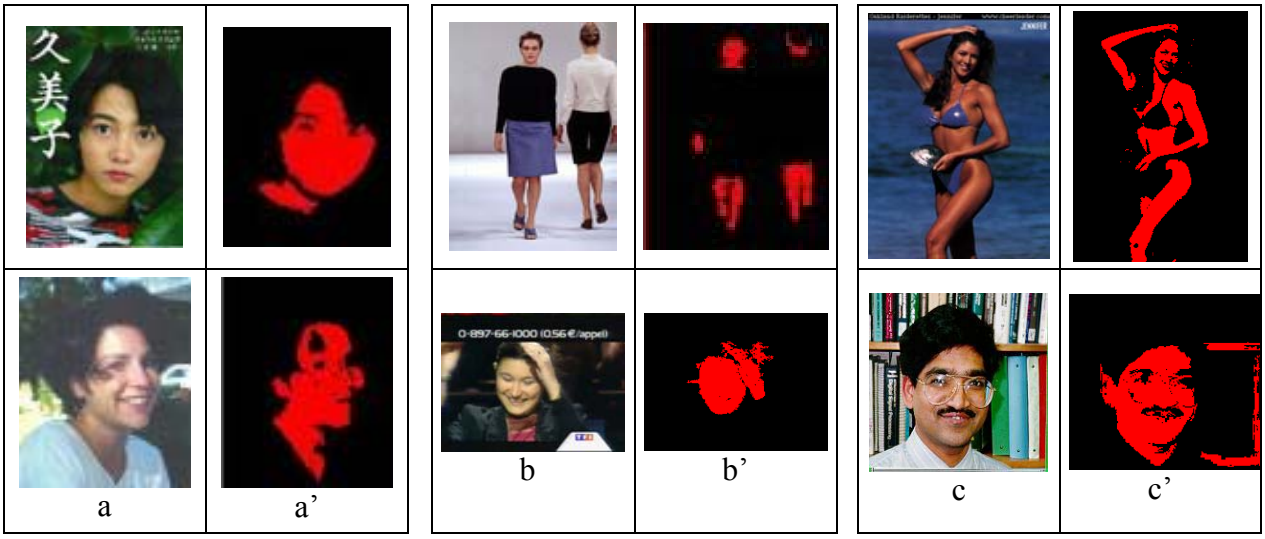


Figure 4. 29. *Résultats de détection et de segmentation en régions de peau*

Les résultats présentés dans la figure 4.29 sont issus du pipeline d'identification puis de segmentation d'une image cible en régions de peau. Les résultats sont prometteurs, néanmoins, nous pouvons signaler quelques problèmes. Dans certains cas, notre modèle ne détecte pas la totalité d'une région de peau et par conséquent nous nous trouvons avec des régions incomplètes (exemple figure a'). Ce problème est dû essentiellement aux conditions d'éclairage. Dans d'autres cas, notre modèle identifie des pixels de non-peau comme des pixels de peau. Nous avons constaté des cas de fausses détections en présence des pixels de sable ou de bois qui ont une couleur similaire à celle de la peau (figure c'). Ces régions sont généralement filtrées après le processus de segmentation et d'élimination des régions de petites tailles. On trouve aussi certaines régions appartenant à la classe des faux positifs. Ces régions de taille négligeable, sont parfois connectés à des régions de peau et donc ne peuvent être éliminées. La figure b' montre une petite région connectée à la main de la dame, qui n'a pas été supprimé après segmentation.

Nous signalons également qu'une variation du seuil λ peut être utile dans certaines applications. Notamment pour la classification d'images en images adultes/images non adultes où l'augmentation du seuil λ permet d'améliorer la performance du système de classification. En effet, dans une image adulte la taille des régions de peau est grande. Un autre exemple d'application où on peut augmenter légèrement le seuil est celui de la classification de portraits d'une personne en gros plan, plan américain et plan en pied où les tailles des régions significatives sont importantes. Dans l'image c, une augmentation du seuil va éliminer les fausses détections correspondant à une région de bois pour ne considérer que les régions significatives pour la classification. La figure 4.30 illustre cette amélioration.

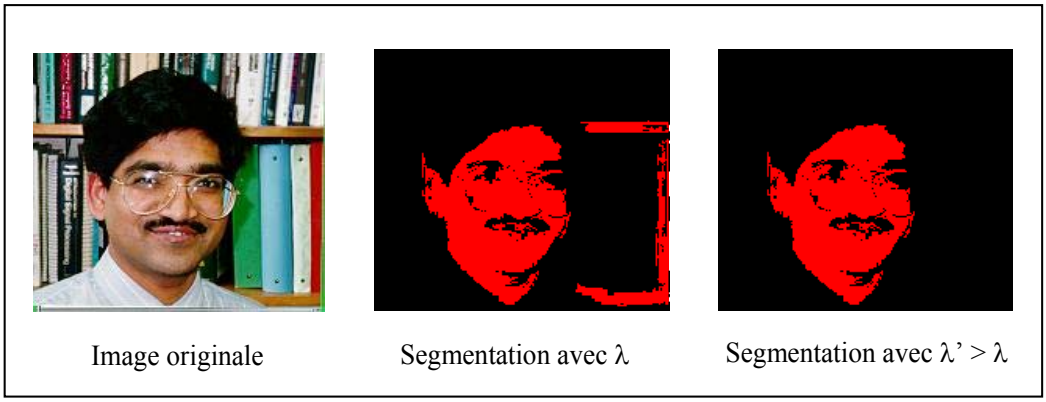


Figure 4. 30. *Résultat de segmentation après augmentation de* λ

Enfin, nous avons suivi l'intégration de notre modèle de couleur de peau dans un système de détection de visage élaboré au sein de notre équipe (cf. figure 4.31). Les expérimentations sur des séquences vidéo, capturées et enregistrées en locale montrent l'efficacité de notre modèle par rapport à celui de Compaq par une amélioration significative du taux de détection de visages. Le tableau 4.9 compare les résultats obtenus par notre modèle de couleur de peau (B) et celui de compact (C). Le paramètre qui a été évalué entre les deux modèles est le nombre total de personnes détectées par heure de vidéo.

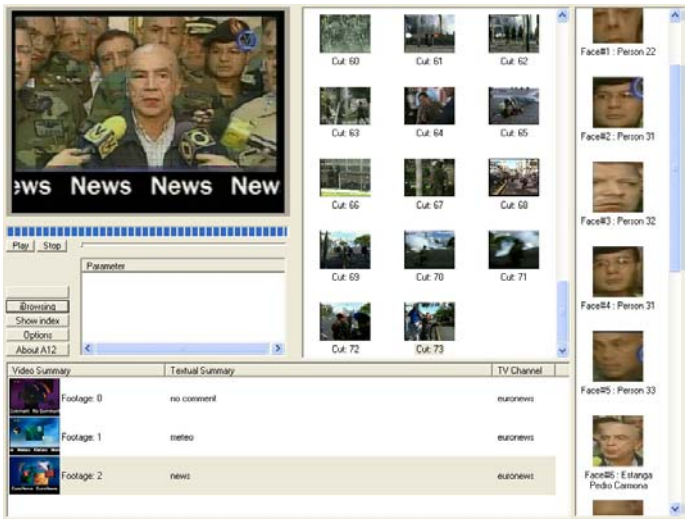


Figure 4. 31. *Détecteur de visage dans la vidéo*

Tableau 4. 9. *Détection de visage dans la vidéo*

Conditions d'expérimentation :	Vidéo capturée de différentes chaines télé du monde (en total 10 chaines)
Durée	15 heures
Intervalle de temps	1 mois
Détection correcte de visages	B – 64 (C – 60) différentes personnes par heure
Fausse détection de visages	6 sujets par heures

Performance	Moyenne fps – 100 (Intel Pentium 2.0 GH)
Taille minimum	20x20 pixels
Taille Maximum	infinie
Liberté de rotation	Jusqu'à 25 degrés

Conditions d'expérimentation :	Programme de la télé française (TF1, France2, France3, M6)
Durée	30 heures
Intervalle de temps	15 jours
Détection correcte de visages	A – 29 (B – 26) différentes personnes par heure, A – 460 (B - 410) visage par heure
Fausse détection de visages	6 sujets par heure
Performance	Moyenne fps – 100 (Intel Pentium 2.0 GH)
Taille minimum	20x20 pixels
Taille Maximum	infinie
Liberté de rotation	Jusqu'a 25 degrés

Les résultats illustrent la fiabilité de notre modèle pour une telle application.

4.8 Application : classification des portraits

Nous avons testé notre modèle de peau sur deux applications : la première est celle de la classification d'un portrait en gros plan, plan américain et plan en pied, que nous présentons dans cette section ; tandis que la deuxième application porte sur le filtrage de sites adultes sur Internet qui sera présenté dans le chapitre suivant.

La définition du cadrage d'un portrait consiste en la classification de celui-ci en gros plan, plan américain ou plan en pied (cf. figure 4.32). Nous avons entrepris ce travail dans le cadre du projet RNTL Muse visant à mettre au point un moteur de recherche multimédia sur le Web. Ce projet associe des laboratoires de recherche comme Prisme de l'Université de Versailles, SIS de l'Université de Toulon, LIRIS ECL avec une start-up XML-media et une agence de photo Editing. Nous avions la responsabilité de développer un moteur de recherche sur l'image [9].

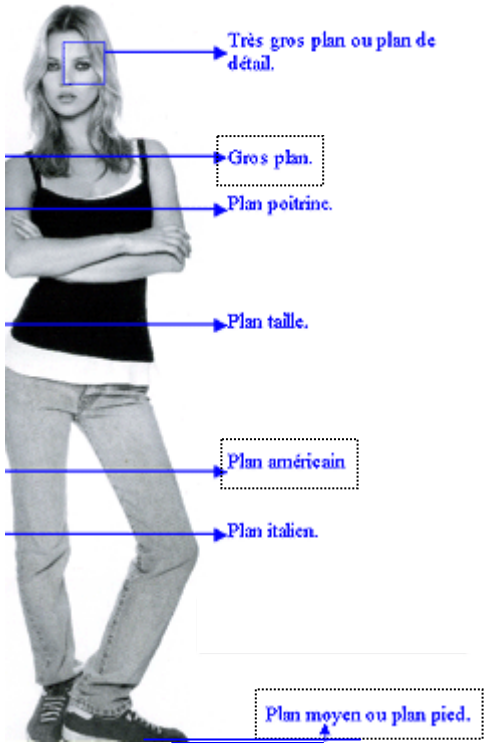


Figure 4. 32. *Valeurs des cadres sur un personnage (Classification des valeurs de plan)*

Nous donnons dans la suite des brèves définitions des différents plans auxquels on s'intéresse :

- Le gros plan : il peut varier dans la découpe inférieure entre la ligne des épaules (clavicules) et le dessus de la poitrine, la limite supérieure du gros plan comprend tout ou partie des cheveux, sans espace au-dessus de la tête.
- Le plan américain : il donne un peu de recul vis a vis de la personne photographiée, en intégrant dans l'image les bras et le premier tiers des cuisses.
- Le plan en pied : l'ensemble de la personne figure dans l'image. Ce plan ne tolère aucune coupe, la présentation de l'intégralité de l'individu permet de lui adjoindre des éléments de décor, tout comme le cadrage en plan américain.

Ce genre de classification peut nous servir pour répondre à des requêtes de type Personne et cadrage, par exemple « Chirac en gros plan ». Ainsi on aura un résultat plus raffiné par rapport à une requête exprimée par un simple mot comme « Chirac ». Cette classification a aussi été utilisée pour la classification d'une vue en gros plan par rapport à une vue générale où on a une vue sur tout le court dans un match de tennis.

Cette définition du cadrage repose essentiellement sur notre modèle de peau et une segmentation de régions de peau dans une image. Cette dernière nous permet de calculer le pourcentage de peau dans l'image, et d'en extraire des indices spatiaux sur l'emplacement de chaque région. Ensuite nous procédons par un filtrage de régions en ne gardant que les régions significatives pour notre définition du cadrage. Enfin un apprentissage supervisé permet de produire les règles de prédiction adéquates à la classification de portraits. Le pseudo algorithme est le suivant :

Algorithme 4.5 : Algorithme de classification de portraits

Entrée : Image RGB (portrait d'une personne)

1. Redimensionner l'image
2. Image dans l'espace HSV
3. Segmentation de l'image en régions de peau
4. Détection des régions significatives (emplacement par rapport à l'axe centrale de l'image)
5. Calcul du pourcentage de peau dans ces régions par rapport à l'image entière

Résultat : Gros plan/ Plan Américain / Plan en pied

La figure 4.33 présente les différentes étapes d'identification des régions d'intérêt de peau

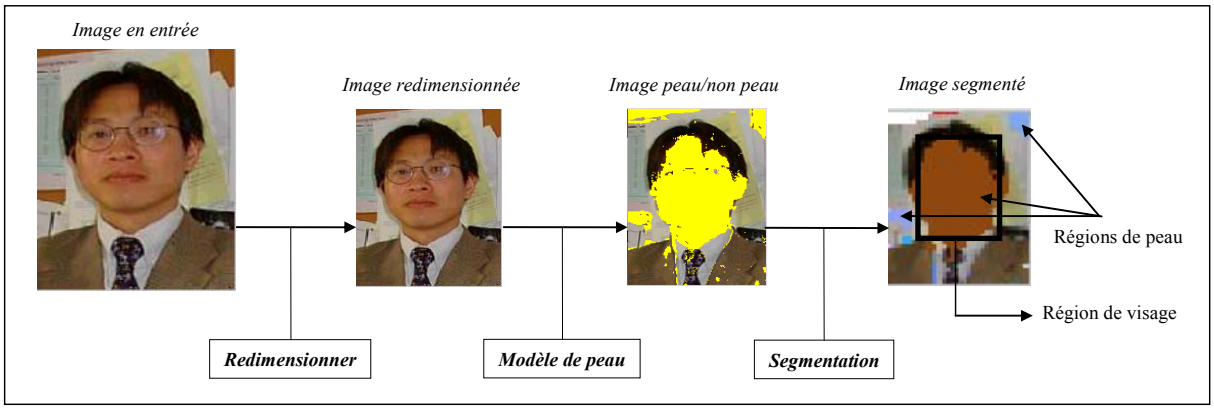


Figure 4. 33. *Processus de segmentation*

4.8.1 Identification des régions de peau significatives

Après avoir segmenté une image en régions de peau homogènes, il est nécessaire de localiser les régions de peau les plus significatives pour notre classification. Pour ce faire nous calculons pour chaque région son centre de gravité et sa taille par rapport à l'image. Par la suite, nous éliminons les régions de taille négligeable et les régions dont la distance de leurs centres de gravité par rapport à l'axe central de l'image dépasse un seuil β.

- Calcul du Centre de gravité :

Etant donné N pixels (x_0,y_0), (x_1,y_1), … (x_{N-1},y_{N-1}) qui composent une région homogène R, le centre de gravité G, de coordonnée (x_G, y_G) est défini par:

$$x_G = \frac{1}{N}\sum_{i=0}^{N-1} x_i \quad \text{Et} \quad y_G = \frac{1}{N}\sum_{i=0}^{N-1} y_i \qquad (4.\ 11)$$

- Critère de sélection :

Une région de peau est considérée comme significative si la distance de son centre de gravité par rapport à l'axe central de l'image divisée par la largeur de l'image est inférieure à un seuil β (cf. figure 4.34). La formule de sélection est la suivante :

$$\frac{\|x_g - x_c\|}{W} \leq \beta \qquad (4.\ 12)$$

Avec :

- x_g : l'abscisse du centre de gravité de la région en question
- x_c : représente l'axe central de l'image
- W : la largeur de l'image

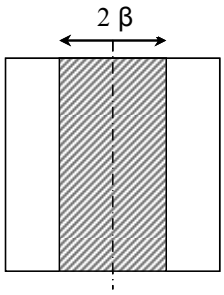


Figure 4. 34. *Zones significatives*

Le seuil β est obtenu suite à un apprentissage supervisé sur une base de 100 images. Pour chaque région identifiée comme peau dans ces images, nous avons calculé son centre de gravité et sa distance par rapport à l'axe central divisée par la largeur de l'image. Par la suite une classification manuelle de ces régions en régions significatives/non significatives a été faîte. Enfin la phase d'apprentissage nous a permis de fixer le seuil β à 0.25.

La figure 4.35 montre les résultats obtenus après une segmentation et une sélection des régions d'intérêts dans des images de portraits.

Les régions colorées en marron sont les régions choisies pour calculer le pourcentage de peau effective dans l'image. Les autres régions (avec d'autres couleurs) sont des régions de peau dont les distances de leurs centres de gravité dépasse le seuil β.

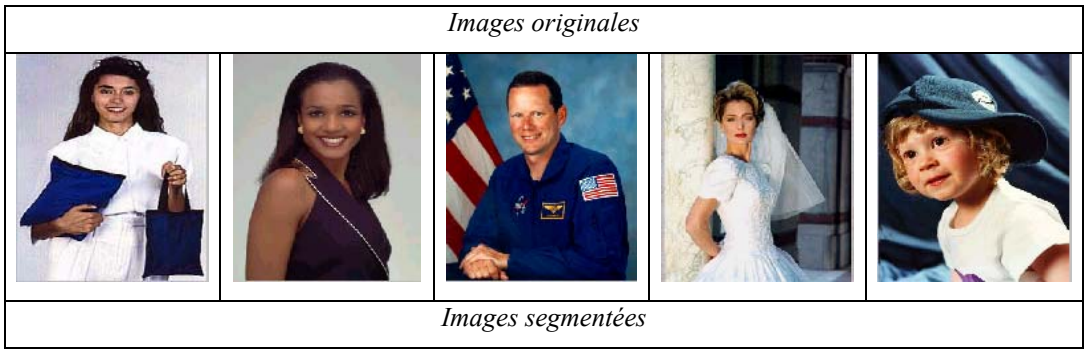


Figure 4. 35. ***Sélection des régions significatives***

4.8.2 Extraction des règles de prédiction :

Afin d'établir un modèle de prédiction permettant de déterminer le cadrage d'une image à partir du pourcentage de peau dans celle-ci nous avons procédé comme suit :

- Choix d'un ensemble d'image (ensemble d'apprentissage)
- Calcul de pourcentage de peau pour chaque image
- Classement manuel des images en gros plan, plan américain et plan en pied
- Apprentissage supervisé pour produire des règles liant les descripteurs à ces classifications de haut niveau.

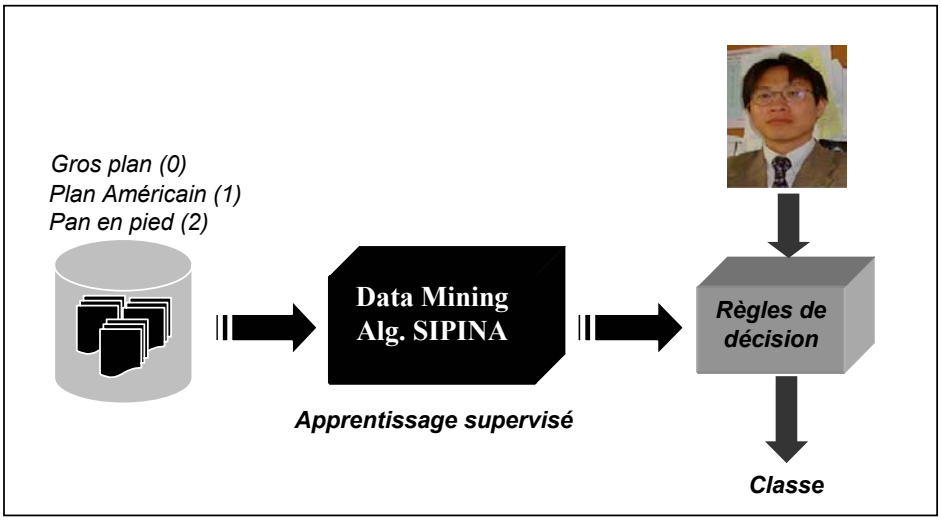


Figure 4. 36. *Data mining pour l'extraction des règles de décision*

Le modèle de prédiction obtenu permet, pour une image de portrait pour laquelle nous ne connaissons pas sa classe mais dont nous pouvons calculer son pourcentage de peau, de prédire sa classe qui peut être gros plan, plan américain ou plan en pied (cf. figure 4.36).

4.8.3 Résultats

Nous avons testé notre méthode sur un corpus constitué de 435 images caractérisées par la complexité de leurs décors. Les résultats obtenus sont encourageants : nous avons atteint un taux de réussite de 90 %. La figure 4.37 présente l'interface de notre application, alors que la figure 4.38 présente quelques résultats.

Figure 4. 37. *Interface graphique de l'application*

Image originale	*Image redimensionnée*	*Image segmentée*	*Class*
			GROS PLAN
			AAMERICAIN
			EN PIED

Figure 4. 38. *Exemples de résultats*

4.9 Conclusion

Nous avons présenté dans ce chapitre nos travaux pour l'élaboration d'un modèle de peau qui permet de détecter des régions de couleur de peau dans une image en fonction d'un certain nombre de critères précis. Cette étude a mis en évidence l'intérêt des outils de data mining pour extraire les informations les plus pertinentes à partir de données complexes et diverses. Ces outils nous ont permis d'extraire des connaissances qui ont abouti à la construction d'un modèle de prédiction de peau/non peau robuste par rapport à la variabilité des conditions de luminosité et à la richesse des ethnies. Les résultats de nos expérimentations montrent que ce processus de fouille de données apporte un plus indéniable. En effet, une étude comparative

avec d'autres méthodes montre un gain d'efficacité substantiel, surtout lorsque la méthode est combinée avec une méthode de segmentation appropriée [2]. Ce modèle de peau a d'abord été appliqué avec succès dans deux applications : la détection de visages dans les séquences vidéo [10] et la définition du cadrage du portrait [9]. Dans le chapitre suivant, nous étudions en détail une autre application du modèle de peau à un problème important : le filtrage de sites pornographiques. Nous y verrons que la combinaison de l'analyse sur des contenus textuels, structurels et visuels par des techniques de data mining nous a permis d'aboutir à un système de filtrage WebGuard très efficace.

Chapitre 5

Filtrage des sites sur Internet

5.1 Introduction

De nos jours, Internet prend une place grandissante dans la vie quotidienne et dans le monde professionnel. Le public qui y a accès est de plus en plus large, mais aussi de plus en plus jeune. Les enfants trouvent chaque jour un accès plus facile à la toile. Cet accès de plus en plus large ne va pas sans inconvénients, les sites à caractère adulte, violent, raciste, etc. en sont un. En effet, ces sites sont souvent en accès libre, ce qui pose un problème évident vis à vis des enfants. Le problème est aussi récurrent dans le monde de l'entreprise puisque de nombreuses personnes abusent de leur connexion professionnelle pour naviguer sur ce genre de sites.

Comme pour toute invention qui modifie le cours de la vie, Internet et ses services offrent la plate-forme idéale aux personnes qui ont les intentions les moins honorables. Tout le monde montre les sites à caractère pornographique du doigt. C'est en effet le premier type de contenu à être diffusé sur les sites à contenu préjudiciable, notamment pour les enfants. Selon l'Institut Jupiter MMXI, la vente de contenus et services "pour adultes", c'est à dire à connotation érotique ou pornographique, aurait représenté en 2001/2002 plus de 70% des revenus de la vente de contenus numériques auprès du grand public dans le monde. Ce chiffre pourrait se maintenir en 2005/2006, en tête des ventes de contenus en ligne. Contre toute attente, une étude réalisée en 1986 aux USA a révélé que l'une des tranches visées par les pornographes concerne les enfants âgés entre douze et dix-huit ans [175]. Quoi de plus choquant que de réaliser que des sociétés spécialisées dans la pornographie visent des enfants ! Fort heureusement, il existe des lois qui réglementent la vente des photographies et des magazines à caractère pornographique. Mais qu'en est-il sur Internet ?

D'après une étude réalisée en mai 2000, 60% des parents sont inquiets lorsque leurs enfants naviguent sur Internet, en particulier à cause de la pornographie [175]. Ce problème concerne autant les parents que les sociétés. Ainsi, la société Rank Xerox a licencié en octobre 1999 quarante employés qui naviguaient sur des sites pornographiques pendant leurs heures de travail. Pour éviter ce genre d'abus, la société a installé des logiciels qui permettent de surveiller le comportement des employés sur Internet.

Pour faire face à ce fléau, il existe un ensemble de produits commerciaux sur le marché qui proposent des solutions de filtrage de sites adultes. Un nombre significatif de ces produits se fondent sur une liste de mots clés peu convenable pour les enfants (nudité, pornographique, sexe, etc.) Ils détiennent également une liste noire de sites Web connus pour leur manque total de pudeur et dont ils interdisent l'accès. Cette liste est construite la plupart du temps manuellement et est nullement basée sur une méthode automatique de classification. Mais, comme nous savons, l'information sur le Web est fortement dynamique. Chaque jour, beaucoup de sites apparaissent alors que d'autres disparaissent. En plus le contenu des sites et surtout les liens sont mis à jours fréquemment. De ce fait, les systèmes manuels de classification et de filtrage sont impraticables et inefficaces. La nature du Web réclame de nouvelles techniques pour classifier et filtrer les sites Web inappropriés.

La classification automatique des sites adultes est un exemple représentatif d'un problème général de catégorisation des sites Web car elle combine généralement le contenu textuel avec

le contenu visuel de la page. Plusieurs travaux de recherche sur la classification et la catégorisation des documents Web ont montré d'une part qu'une classification basée seulement sur le contenu textuel n'est pas performante, et d'autre part que les descripteurs basés sur le contenu structurel comme les hyperliens et les documents voisins de la page aident considérablement à améliorer le taux de la classification [176] [191].

Dans ce chapitre, nous présentons notre solution de classification et de filtrage de sites adultes sur Internet. À la différence de la majorité des logiciels commerciaux, qui sont principalement basés sur la détection des mots interdits ou sur une liste noire, collectés et mis à jour manuellement, notre solution, baptisée WebGuard, réalise la classification et le filtrage de sites à caractère pornographique par un apprentissage qui s'appuie sur une combinaison judicieuse de plusieurs algorithmes de data mining avec non seulement une analyse du contenu textuel mais aussi du contenu structurel et visuel.

Expérimenté sur une base de test de 400 sites composés de 200 sites adultes et 200 non adultes, WebGuard affiche un taux de classification de 97,4%. D'autres expériences sur une liste noire de 12 311 sites adultes, manuellement rassemblés et classifiés par le ministère de l'éducation français, montrent que WebGuard atteint un taux de classification de 95,62%.

Dans ce chapitre, après une revue de littérature sur les travaux qui ont porté sur le filtrage de sites, faisant l'objet de la section 2, nous décrivons dans la section 3 l'architecture et le principe de fonctionnement de notre solution WebGuard. Dans la section 4, nous présentons et argumentons les critères textuels et structurels utilisés par notre technique. L'intégration du contenu visuel basé sur la couleur de peau est développée dans la section 5. Par des expérimentations significatives, la section 6 évalue, valide et compare avec les logiciels du marché notre approche de classification et de filtrage de sites adultes. La section 7 présente quelques issus d'implémentation. Une synthèse de nos contributions est décrite dans la section 8 en guise de conclusion du chapitre.

5.2 Etat de l'art et étude de la concurrence

De nombreux travaux de recherche dans la littérature ont déjà montré l'intérêt croissant pour la classification et le filtrage de sites Web dans le but de limiter l'accès et la prolifération des contenus préjudiciables sur la toile. Il existe également une panoplie de produits commerciaux de filtrage sur le marché. Dans cette section, nous décrivons tout d'abord notre base de test que nous appelons « Base de test MYL » et qui a servi pour évaluer et comparer divers travaux de recherches et produits commerciaux. Nous étudions ensuite quelques travaux de recherches significatifs dans ce contexte. Enfin, nous évaluons sur notre base de test MYL les performances des logiciels existants les plus connus sur le marché et en tirons les conclusions.

5.2.1 Base de test MYL

Afin d'avoir une base de test représentative, nous avons collecté manuellement 400 sites : 200 à caractère adulte et 200 à caractère non adulte. Les sites adultes ont été regroupés en deux classes : ceux qui ne comportent que des images et ceux composés de texte et d'images. Cette séparation a pour but de mettre en évidence le rôle de l'analyse d'images.

Actuellement, pour les sites adultes de notre base de test MYL, nous avons inclus des :

- sites érotiques ;
- sites pornographiques ;
- sites de hack présentant des images à caractère pornographique ;
- sites de jeu inoffensif en journée et présentant textes et images illicites en soirée.

Pour la sélection des sites non adultes, nous avons inclus ceux qui peuvent prêter à confusion, en particulier les sites de santé, de sexologie, de défilé de modes, de lingerie, etc.

5.2.2 Travaux existants

Il existe quatre approches de filtrage de sites adultes : la technologie de l'étiquetage (PICS), le filtrage par liste de sites autorisés ou interdits, le filtrage par mots clé et le filtrage par analyse intelligent du contenu [177].

5.2.2.1 *Technologie de l'étiquetage (PICS)*

- *Le protocole PICS*

En 1995, le World Wide Web Consortium[1](W3C) a élaboré PICS[2] (Platform for Internet Content Selection), un standard de programmation permettant de véhiculer des informations concernant le genre de contenus représentés sur les sites Web. Le standard PICS permet à n'importe quelle organisation de définir son propre système de classification et de donner ainsi la possibilité aux personnes utilisant ce système de bloquer (ou de rechercher) des sites en fonction de leur contenu.

Il est important de comprendre que le standard PICS n'est pas un système de classification, mais une méthode de codage utilisée par les systèmes de classification. PICS propose aux éditeurs de sites Internet une méthode normalisée pour décrire le contenu de leurs pages. Les éditeurs de sites peuvent ainsi, de leur propre gré, attribuer une cote à leurs pages.

Les navigateurs Netscape ou Internet Explorer depuis leur version 4.5 sont capables de lire ce genre de description. Il appartient donc à l'utilisateur d'activer la fonction de filtrage, de choisir l'un ou plusieurs de ces systèmes de classification et de déterminer les niveaux qu'il estime acceptables pour chacun des critères.

Les systèmes d'étiquetage les plus connus à l'heure actuelle sont ceux qui sont proposés par le RSACi[3] (Recreational Software Advisory Council) et SafeSurf[4]. Tous les deux sont basés sur le protocole PICS (cf. figure 5.1).

[1] http://www.w3.org.
[2] http://www.w3.org/PICS.
[3] http://www.rsac.org.
[4] http://www.safesurf.com.

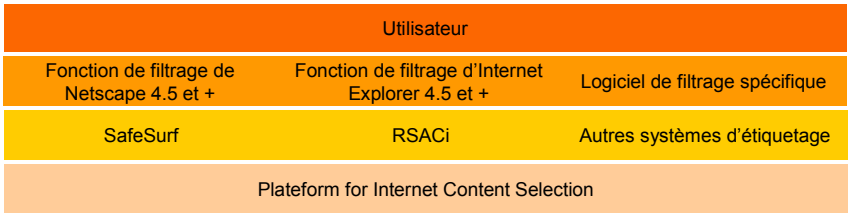


Figure 5. 1. *Superposition en couches des divers éléments constituant un système de filtrage par étiquettes*

Le système RSACi se base sur quatre catégories (violence, nudité, sexe et langage (langage choquant, haineux, etc.)) auxquelles un numéro est associé pour indiquer le degré ou le niveau de contenu potentiellement offensant. Chaque numéro est compris entre 0 (la page ne contient aucun contenu potentiellement offensant) et 4 (la page présente un contenu potentiellement très offensant).

Le système « SafeSurf » est l'un des systèmes de classification PICS les plus détaillés. Les sites Web cotés à l'aide du système SafeSurf identifient leur adéquation à des groupes d'âge spécifique. En outre, les pages Web peuvent contenir jusqu'à dix étiquettes pour décrire leur contenu. Chaque étiquette présente neuf niveaux.

- *Problèmes liés à l'étiquetage des sites*

L'efficacité de PICS dépend très fortement de l'adhésion des concepteurs de sites ou des organismes d'évaluation externes. A l'heure actuelle, peu de sites sont classés étant donné que personne n'oblige actuellement les éditeurs de site à étiqueter leur contenu. Si le choix est fait de refuser tous les sites non classifiés, l'utilisation d'Internet se trouve fortement limitée. De nombreux sites, non classifiés, vont être bloqués lors de la navigation alors qu'ils peuvent correspondre au profil de l'utilisateur. La seconde option est d'accepter l'affichage des sites non étiquetés. Dans ce cas-là, l'utilisateur prend le risque de voir apparaître des pages à contenus inappropriés. Dans les deux cas, PICS devient un instrument inefficace.

En plus, l'auto-évaluation ne peut vraiment s'imposer que si le vocabulaire d'évaluation se standardise au niveau mondial, d'où risque de se retrouver avec un Internet nivelé par la culture dominante. On notera aussi que certains types de contenus ne peuvent pas par nature être étiquetés de manière pertinente : médias d'information, contenus artistiques, sites personnels, forum, *IRC*, etc.[178].

Enfin, un autre argument s'oppose de surcroît à une réelle efficacité de *PICS*, aujourd'hui et dans le futur est que la masse d'information est trop importante pour être classifiée. En effet, Le nombre de sites présents sur Internet ne cessant de croître à une vitesse phénoménale, la possibilité d'évaluer un nombre significatif de sites ne sera donc jamais effective.

5.2.2.2 *Liste de sites autorisés ou interdits*

- *La liste de sites interdits ou liste «noire»*

Une liste noire contient un ensemble de sites, motifs génériques (par exemple toutes les adresses contenant le mot « nue ») ou domaines à exclure de la navigation. C'est donc un

ensemble de sites interdits.

Les techniques basées sur une liste noire permettent de créer un Internet où tout est autorisé sauf la consultation de quelques sites. On garde donc la possibilité de naviguer librement d'un site à un autre, tout en restreignant les risques d'accéder à un site inapproprié.

Cependant, une liste noire ne peut jamais être exhaustive puisque de nouveaux sites apparaissent constamment, les logiciels de filtrage ne pourront, même avec des mises à jour régulières, bloquer la totalité des sites adultes. Chaque liste sera obsolète dès le moment où elle aura été mise sur le marché puisque tout nouveau site apparaissant après la mise à jour ne sera pas contenu sur cette liste et ne sera, par conséquent, pas bloqué. Ce problème a été aussi soulevé par Hochheiser et Aftab[179].

De plus, comme le souligne Breckelmans[180], par souci de simplicité et de rapidité, les éditeurs de listes noires ont tendance à interdire complètement des sites, des domaines ou des adresses entières plutôt que des pages individuelles. Par conséquent, certains serveurs publics qui hébergent une petite proportion de pages érotiques personnelles seront bloqués, alors que 95% de leurs contenus reste tout public.

Il faut savoir également que la manière de sélectionner les sites interdits, ainsi que les sites autorisés est assez controversée. Des collaborateurs mal ou pas du tout formés ainsi qu'un contrôle de qualité déficient peuvent avoir pour conséquence des listes peu fiables soit parce qu'elles bloquent trop de sites ou pas assez.

Les méthodologies et la déontologie mise en œuvre pour mettre une URL dans une liste noire ne sont pas aussi transparentes pour les utilisateurs. En effet, l'utilisateur est incapable de savoir quels sites sont bloqués et pour quelle raison. Ce manque de transparence compromet le rôle de l'utilisateur comme un participant actif et réfléchi dans son utilisation d'Internet.

Nous signalons qu'une structure nationale au niveau interministériel est mise en place afin de coordonner et de centraliser l'offre de liste noire. Les partenaires (Ministère de l'Intérieur, délégation à la famille, MJENR, etc.) contribuent dans leur domaine respectif à améliorer et pérenniser cette liste ainsi que la structure et le suivi institutionnel. Un site de référence (site sur educnet) a été créé afin de regrouper l'ensemble des informations et des moyens mis en place par cette structure. A l'heure actuelle, une liste « noire » est librement téléchargeable sur ce site.

- *La liste de sites acceptables ou liste «blanche»*

Une liste blanche contient l'ensemble des sites autorisés. Toute tentative d'accès à n'importe quel site ne figurant pas sur cette liste sera automatiquement refusée. Il n'a pas vocation à l'universalité mais convient bien également à une utilisation à l'école et dans l'entreprise. Les sites étant sélectionnés pour leur qualité, ces listes pourront être publiées et éviteront les mises à l'index camouflé permis par PICS ou le système de listes noires. La mise à jour des listes blanches est facilitée par l'intérêt que les créateurs de sites auront à se signaler spontanément aux éditeurs de listes blanches, puisqu'il s'agira d'une classification positive et

non d'une mise à l'index, mais le problème c'est que dans ce cas la recherche d'informations se rapproche d'une recherche documentaire.

Il arrive souvent que certains sites à contenu tout à fait licite disparaissent, et leurs adresses récupérées par des sites pornographiques d'où la nécessité d'une vérification régulière par l'administrateur.

5.2.2.3 *Filtrage par mots clés*

En complément aux listes « noire » et « blanche » les logiciels de filtrage peuvent également effectuer un contrôle des contenus par mots clés ou phrases clés. À l'aide d'un outil d'analyse de texte, le programme vérifie tous les mots de la page avant que celle-ci ne s'affiche. Si un mot « interdit » est décelé, que ce soit dans une page Web, dans le titre d'un groupe de discussion ou dans celui d'un forum de dialogue en direct, le logiciel de filtrage bloquera l'affichage de ces données. Selon Hochheiser [181], l'un des problèmes avec ce genre de recherche est que seuls des mots bruts et décontextualisés sont recherchés, une telle recherche n'est pas capable par exemple de faire la différence entre « fortes poitrines » et «cancer de la poitrine ». Dans ce contexte quelques associations américaines comme Peacefire [182] et NCAC [183] ont testé des outils qui se basent sur cette approche et toutes s'accordent pour dire que cette technique n'est pas en mesure de filtrer efficacement le Web. Les outils bloquent des pages qui n'ont pas lieu de l'être et en laissent passer d'autres au contenu explicite. En décembre 2000, Peacefire publiait ainsi la liste de trente sites bloqués par les logiciels de filtrage et qui, pourtant, étaient consacrés aux droits de l'homme, tel celui d'Amnesty International.

Le second problème souligné par Hochheiser est l'impossibilité, à l'heure actuelle, d'analyser les représentations graphiques, ce qui implique que des images «sexuellement explicites» ne seront bloquées que si le texte qui les accompagne contient un ou plusieurs mots se trouvant sur la liste des mots clés « interdits ».

Ce système se heurte généralement au problème de la langue : une même expression devant être enregistrée dans toutes les langues véhiculées sur Internet. Par ailleurs les éditeurs de sites adultes déploient des trésors d'imagination pour trouver des orthographes de substitution, par exemple en écrivant S.E.X.E au lieu de sexe.

5.2.2.4 *Filtrage par analyse intelligente du contenu Web*

La classification des sites adultes par une analyse intelligente du contenu Web s'intègre dans une problématique plus générale, celle des systèmes automatiques de classification et de catégorisation de sites Web. La réalisation de tels systèmes doit s'appuyer sur un processus d'apprentissage automatique et plus précisément sur un apprentissage supervisé. Par exemple, Glover et al.[191] ont utilisé les SVM (Support Vecteur Machine) pour produire leur classifieur de documents Web. Tandis que Lee et al.[177] ont utilisé les réseaux neuronaux pour élaborer leur solution de filtrage du contenu Web. Les SVM ont montré leur efficacité dans de nombreuses applications de classification, leur problème étant la difficulté de trouver des fonctions de noyau appropriées permettant de plonger dans un espace de haute dimension des données d'apprentissage des deux classes non linéairement séparables. En contre partie,

les réseaux neuronaux, malgré leur efficacité face à des données non linéairement séparables, ne fournissent pas des résultats facilement interprétables.

La construction du vecteur de caractéristiques est un problème fondamental de l'apprentissage automatique. Généralement, le vecteur de caractéristique est lié à une connaissance a priori du problème de classification. Plus le vecteur est bien choisi, meilleure est la performance du classifieur. Il est réputé que la classification des documents Web est difficile [176]. En effet, si un classifieur basé sur une analyse textuelle peut atteindre un taux de classification compris entre 80% et 87% [184] sur un corpus homogène tels que les articles financiers, il est de notoriété qu'il est inadéquat pour des documents Web caractérisés par la complexité de leurs structures et la diversité de leur contenu qui est de plus en plus multimédia.

Afin de prédire la classe adulte ou non adulte d'un site Web, Lee et al. [177] se sont basés dans leur classifieur sur les fréquences des mots indicatifs qui apparaissent dans le site. Cependant, ils ont explicitement exclu les URLs de leur vecteur de caractéristiques en expliquant qu'elles n'apportent pas d'informations supplémentaires pour le processus de classification.

Cependant, contrairement à Lee et al., beaucoup de travaux ont plutôt souligné l'importance de la structure d'une page Web et particulièrement des liens hypertextes. Il a été montré que l'information structurelle d'une page Web est intéressante pour améliorer la qualité des résultats obtenus par un moteur de recherche [185], renforcer la performance d'un robot Web [186], découvrir des communautés du Web [187] et classifier les pages Web [188][189][190][191]. Par exemple, Flake et al. [184] ont étudié le problème d'identification de la communauté du Web en se basant seulement sur des hyperliens. Ils considèrent qu'un hyperlien entre deux pages Web est un indicateur explicite du lien sémantique entre les deux pages. À partir de cette hypothèse, ils ont étudié plusieurs méthodes et mesures, telles que la référence bibliographique couplée, le couplage de co-citation, l'autorité, etc. Glover et al. [191] ont également étudié l'utilisation de la structure du Web pour décrire et classifier les pages. Ils ont conclu que le texte référencé dans le document cible, s'il est disponible, est souvent plus discriminant et plus descriptif que celui du document cible lui même. Ils ont aussi souligné l'importance du texte d'appel, qui résume le contenu d'un hyperlien. Les auteurs ont également prouvé l'importance des mots clés pour la classification des pages Web.

5.2.3 Etude et analyse des logiciels existants

Pour compléter notre étude sur les travaux existants, nous avons également testé un ensemble de six produits commerciaux de filtrage existants sur notre base de test MYL. Ces tests ont pour but non seulement de comparer les performances de ces logiciels avec le notre mais aussi, et surtout, de mettre en évidence les cas pathologiques à prendre en compte pour obtenir le filtrage le plus fin possible.

Les six logiciels de filtrage que nous avons testé sont les suivants :

- Microsoft Internet Explorer (RSACi)[192] ;
- Cybersitter 2002[193];

- Net Nanny 4.0.4[194];
- Norton Internet Security 2003 [195];
- PureSight Home 2.6 [196];
- Cyber Patrol 5.0 [197].

Les résultats de ces différents tests sont décrits dans la figure 5.2.

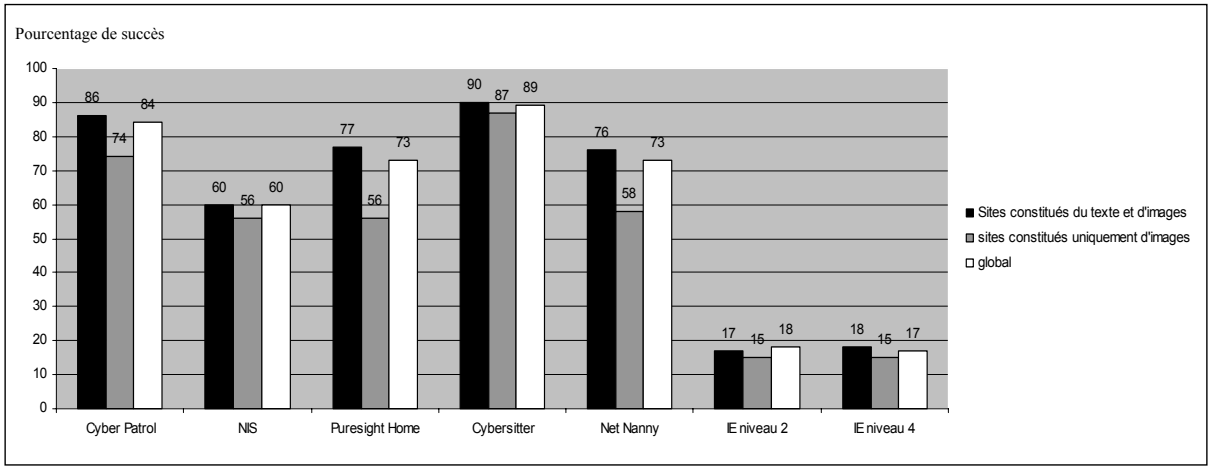


Figure 5. 2. *Taux de succès de classification de six produits commerciaux sur la base de test MYL*

Comme le montre la figure 5.2, le taux de classification peut atteindre 90% pour le meilleur produit. Nous signalons qu'une étude indépendante de la nôtre sur 10 systèmes commerciaux de filtrage, parmi les plus populaires, a été effectuée sur une base composée de 200 pages adultes et 300 pages non adultes. Cette étude a fourni des conclusions similaires au niveau de la performance [177].

En plus des inconvénients que nous avons décrits dans la section précédente, ces tests nous ont permis également de mettre en évidence plusieurs problèmes auxquels nous devrons faire face dans la classification et filtrage de sites adultes :

- *Niveau de réglage :* La plupart des logiciels reconnaissent les sites à caractère fortement pornographique et en avertissent l'utilisateur. En revanche, les sites érotiques échappent à la totalité des logiciels. Dans le même ordre d'idée, il existe un inconvénient relativement fort, présent sur tous les logiciels étudiés hors mis Cyber Patrol 5.0, ce problème se manifeste par un niveau de filtrage difficile à régler. La vision est souvent manichéenne : soit tout passe, soit rien ne passe. Les degrés de filtrage sont souvent là sans réelle utilité car leur fonctionnement est aléatoire pour ne pas dire sans résultat. Par exemple, Microsoft Internet Explorer propose plusieurs réglages qui sont complètement décorrélés entre eux. Pour citer un exemple, prenons les réglages nudité et sexe. Si on impose un niveau 0 en sexe pour un niveau maximum de blocage en nudité, on peut quasiment tout afficher. A l'opposé, si on impose un niveau de blocage maximum en sexe pour un niveau 0 en nudité, plus rien ne s'affiche même pas Google.

- *Dépendance vis-à-vis de la langue :* La totalité des logiciels présents à l'heure actuelle sur le marché fonctionnent sur une analyse de texte, avec tous les problèmes qui en découlent.

En effet, qui dit analyse de texte, dit dictionnaire de mots interdits et donc grande dépendance vis-à-vis de la langue utilisée. Les reconnaissances de textes sont souvent effectuées dans une unique langue id est celle de la nationalité du logiciel. C'est ainsi qu'un logiciel américain ne voit pas le caractère pornographique d'un site écrit en français ou en espagnol et vice versa. En utilisant les logiciels américains on a pu facilement accéder à des sites pornographiques dont le lien est en français sans la moindre difficulté.

- *Détection sans filtrage de sites adultes* : Nous avons remarqué qu'il faut distinguer détection et filtrage. En effet, la plupart des logiciels avertissent l'utilisateur du caractère adulte d'un site, mais le blocage est relatif. Ainsi, il suffit bien souvent de cliquer sur OK plusieurs fois après un message du genre: Ce site est à caractère pornographique. Veuillez contacter l'administrateur du programme avant de poursuivre", pour accéder sans problème à la page demandée.

- *Problème lié aux images* : Plusieurs sites pornographiques, érotiques, etc. pour parer à ces logiciels de filtrage utilisent une méthode simple qui consiste à placer le texte dans les images. En effet les Webmasters de sites adultes connaissent le fonctionnement de ces logiciels de filtrage et savent comment les contourner : en mettant le texte dans des images. Nous en avons conclu que l'analyse d'images reste indispensable pour obtenir un fonctionnement optimal d'un logiciel de filtrage.

- *Sites à double face* : Un autre problème mais qui n'est pas des moindres, les sites à "double face". Nous pouvons citer www.distrigame.com en exemple ou encore www.caramail.lycos.fr. Ces sites sont en journée accessibles pour tous. Par contre en soirée ils se transforment en panneau d'affichage pour des publicités à caractère pornographique. Là encore la reconnaissance d'images et le filtrage des pop-up deviennent indispensables. En effet ce genre de site ne peut être géré par une liste blanche ou noire. Les deux lui conviennent, tout ne dépend que de l'horaire à laquelle il est consulté.

5.3 Principe et architecture de WebGuard

Le manque de fiabilité des systèmes étudiés et d'autres inconvénients découvert dans notre précédente étude nous ont conduit à concevoir WebGuard, un système de filtrage efficace. Le but global de notre système est de rendre l'accès à Internet plus sûr en bloquant les sites Web ayant un contenu pornographique tout en conservant l'accès aux sites inoffensifs. Dans cette section, nous présentons d'abord le principe de base de WebGuard. Ensuite, nous posons le problème de classification des sites dans le cadre de l'extraction de connaissances à partir des données.

5.3.1 Principe général de WebGuard

Etant donné la nature dynamique et la quantité énorme des documents Web, nous avons opté pour un système automatique de détection du contenu pornographique. Notre système utilise une approche d'apprentissage supervisé sur un ensemble de sites classés manuellement, afin de produire un modèle de prédiction.

La sélection des caractéristiques appropriées est l'étape la plus importante dans un

processus de data mining. Dans cette étape, il est intéressant de recueillir les intuitions et les connaissances acquises suite à l'étude de l'état de l'art et celle des tests. Ces éléments vont orienter le processus de découverte pour identifier les variables les plus pertinentes et qui sont susceptibles d'expliquer notre problème de classification. De ce fait, nous avons décidé de ne pas limiter notre étude au contenu textuel d'une page Web, mais de l'étendre à la structure de cette dernière décrite par des balises HTML. En plus, compte tenu de l'importance des images dans les documents Web, en particulier pour les sites adultes, un système de filtrage efficace doit inclure une analyse du contenu visuel. L'analyse du texte consiste à parcourir le fichier mot par mot, et à vérifier si ces mots appartiennent à un dictionnaire de mots « interdits ». L'analyse des balises est faîte en fonction de leurs types (lien, image, mots clefs) et fournissent un certain nombre de critères pertinents (liens vers des sites connus, images au nom explicite, mots clefs explicites etc.) Ces critères sont utilisés par la suite dans une phase d'apprentissage pour déterminer le modèle de prédiction. Alors que l'analyse des images est réalisée en se basant sur notre modèle de peau.

Afin d'accélérer la navigation, nous avons choisi de mettre en oeuvre une liste noire qui sera créée est mise à jour d'une façon automatique. Nous avons également décidé d'employer un dictionnaire de mots clés car l'occurrence des mots interdits dans une page Web est un indice significatif de la nature du contenu de cette dernière. Ce dictionnaire est en fait un fichier texte contenant des mots explicitement choquants, et il peut être complété ou changé pour améliorer l'efficacité et étendre le champ d'application à d'autres types de pages Web.

D'un point de vue d'architecture système, WebGuard s'exécute automatiquement lors de l'ouverture d'un navigateur et tourne en tâche de fond. Il doit agir à chaque demande d'une URL (c'est à dire chaque fois qu'une requête HTTP est lancé), et effectuer les actions suivants (cf. algorithme 5.1) :

- récupérer le code source HTML de la page demandée ;
- vérifier si l'URL appartient à une liste noire ou blanche, et sinon analyser le code ;
- déclarer cette page autorisée ou interdite ;
- afficher ou non la page ;
- mettre à jour la liste noire ou la liste blanche.

Algorithme 5.1 : Algorithme de traitement d'une URL

```
Début
  Récupérer_codeHTML( )   // récupérer le code source HTML de la page
  Si (URL € {liste noire})     // vérifier si l'URL appartient à la liste noire
        bloquer la page,
  Sinon
        Analyser_code( ) : booléen    // vrai : page X ; faux : page sain
        Si (vrai)
              Bloquer la page
              Ajouter_liste noire( )     //mise à jour de la liste noire
        Sinon
              Autoriser_accès( )        //afficher la page
         Fin Si
   Fin Si
Fin
```

La figure 5.3 résume le fonctionnement de notre logiciel.

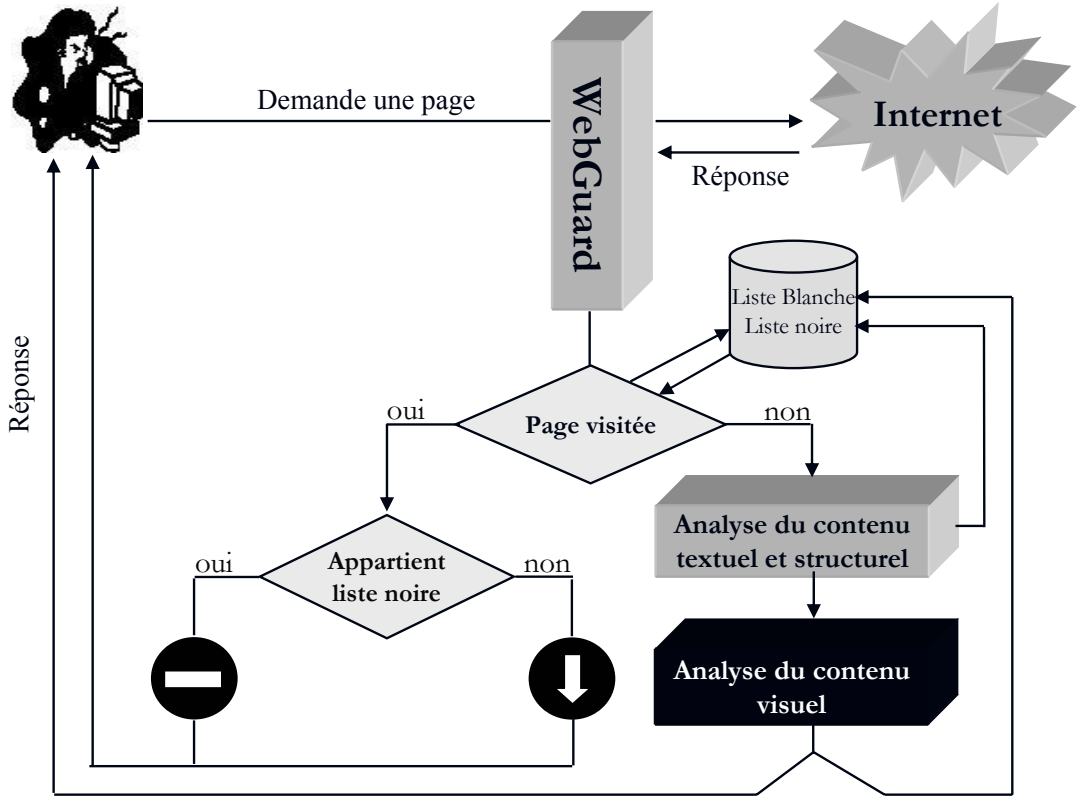


Figure 5. 3. *Schéma de fonctionnement du logiciel WebGuard*

Pour aboutir à une solution de classification et de filtrage de sites adulte, nous avons développé dans un premier temps le cœur du système, c'est à dire le moteur d'analyse des pages Web. Ce dernier repose sur le principe de l'analyse du code HTML d'une page Web et permet de fournir un certains nombre de critères qui ont permis d'alimenter une base d'apprentissage. Dans un deuxième temps nous avons passé à une phase d'apprentissage dans laquelle sont utilisés plusieurs algorithmes de data mining afin de trouver les règles de décision les plus pertinentes, permettant de dire, en connaissant la valeur des critères, si la page est autorisée ou pas. Enfin, nous avons intégré notre analyseur d'images pour affiner et optimiser les résultas.

5.3.2 Utilisation du data mining pour la classification des sites

Afin de bloquer l'accès aux URLs ayant un contenu adulte, nous avons besoin de connaître lesquelles des URLs sont suspectes et lesquelles sont normales. Pour cela nous avons opté pour un apprentissage supervisé sur notre base d'apprentissage, nommé MYL, qu'on décrira en détails dans la section 5.6.1. L'apprentissage supervisé se propose de fournir des outils permettant d'extraire, à partir de l'information dont on dispose sur un échantillon dit d'apprentissage, le modèle de prédiction φ.

Soit S une population de sites concernés par le problème d'apprentissage. A cette population est associé un attribut particulier appelé "attribut classe" noté C. Cet attribut peut

avoir deux valeurs, la valeur 0 si le site est adulte et 1 si le site est normal. A chaque site s peut être associée sa classe C(s).

$$C : S \mapsto \Gamma = \{adulte, non\_adulte\}$$
$$s \to C(s)$$

Dans notre étude nous cherchons un moyen φ pour prédire la classe C. La détermination de ce modèle de prédiction φ est liée à un vecteur de caractéristiques $\vec{X} = (X_i)_{1 \le i \le p}$ que nous avons établit a priori. Ce modèle de prédiction permet, pour un site s issu de S, pour lequel nous ne connaissons pas la classe C(s) mais nous connaissons son vecteur caractéristique, de prédire sa classe.

Le choix des algorithmes de calcul est déterminant pour la performance du modèle. Dans notre cas nous avons utilisé des algorithmes à base d'arbres de décision qui sont ID3, C4.5, SIPINA, Improved C4.5. L'avantage principal de ces techniques est sans conteste la lisibilité du modèle construit. Chaque algorithme produit des règles de type si….alors. La structuration sous forme de règles facilite le travail de validation et de communication du modèle.

La figure 5.4 illustre l'application du principe de data mining au filtrage de sites Web. Nous signalons que les modèles construits sont sensibles à la qualité des données qui leur sont fournies et nous avons été obligé de faire plusieurs itérations qui ont conduit à affiner la recherche et à élaborer de nouvelles variables ce qui nous a permis d'améliorer les résultats obtenus au fur et à mesure des différentes étapes.

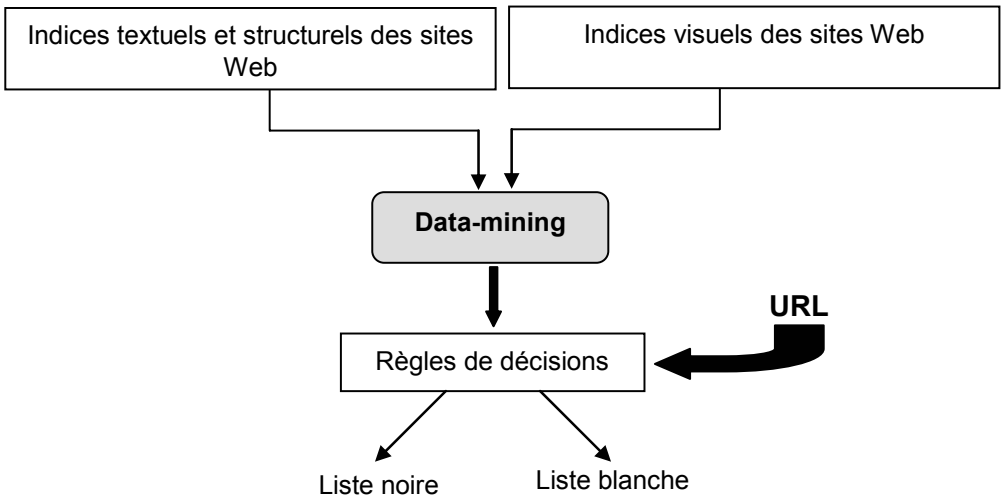


Figure 5. 4. *Application du data mining au filtrage de sites Web*

Afin de valider les modèles obtenus, nous avons évalué ces derniers selon trois méthodes d'évaluations : la méthode des taux d'erreur, la validation croisée et le Bootstrap. Ces évaluations ont été effectuées dans un premier temps sur la base d'apprentissage et dans un deuxième temps sur la base de test. Les détails de ces évaluations seront développés dans la section 5.6 (Expérimentations et recherche du modèle).

5.4 Analyse du contenu textuel et structurel

La sélection des variables pertinentes pour la phase d'apprentissage est une étape principale ayant une influence directe sur la performance du classifieur et par conséquent sur celle du système. Dans cette phase, il est intéressant de recueillir les intuitions et les connaissances acquises suite à notre étude sur les solutions existantes et sur la sélection manuelle des deux bases (la base d'apprentissage et la base de tests). Ceci permettra de déterminer les variables qui discriminent le mieux les pages Web adultes de celles inoffensives.

5.4.1 Variables basées sur le contenu textuel

La fréquence des mots interdits dans une page Web nous semble la variable la plus discriminante. C'est pourquoi nous proposons d'utiliser deux critères, *n_x_mots*, et *%x_mots*, qui présentent respectivement le nombre de mots interdits figurant dans le dictionnaire et leur pourcentage. Cependant, dans une approche basée sur les mots clés, l'efficacité et la qualité d'un classifieur dépendent fortement de la représentativité, de la diversité et de la langue du dictionnaire. Nous avons pris en compte cette propriété, et contrairement à beaucoup de logiciels commerciaux de filtrage, nous avons construit un dictionnaire multilingue comprenant des mots clés courants français, anglais, allemands, espagnols et italiens. Ce dictionnaire compte aujourd'hui 421 mots.

5.4.2 Variables basées sur le contenu structurel

La structure d'une page Web est fondée sur un système de balises (chaînes de caractères délimitées par les symboles < et >) qui décrit leur type (liens hypertexte, images, mots clés, etc.). Glover et al. ont prouvé que l'analyse de cette structure combinée à une analyse textuelle ne peut qu'améliorer la classification et la description de la page Web [191]. Par exemple, il a été montré par Flake et al.[184] que les hyperliens des pages Web sont des indicateurs importants sur les communautés du Web. Nous avons ainsi utilisé le critère *n_xxx_liens* qui compte le nombre de liens répertoriés comme pornographiques (appartenant à la liste noire) et qui permet de décrire le degré d'appartenance de l'URL courante à la liste des URLs suspectes.

Cependant, avant la classification d'une page Web, tous ses hyperliens ne sont pas nécessairement classifiés. Dans la mesure où un hyperlien est composé de deux composantes : une adresse URL et un texte d'appel qui résume son contenu fourni par le créateur de la page, nous analysons aussi les textes d'appel de la même manière que de nombreux moteurs de recherche tel que Google qui permet de renvoyer des pages Web en se basant sur les mots clés qui apparaîtrent dans le texte d'appel. En conséquence, nous avons compté *n_x_liens*, le nombre de liens ayant des mots interdits dans les textes d'appel des hyperliens associés.

De même, intuitivement, une page Web adulte contient beaucoup d'images. Avant de passer à une analyse réelle sur les images, nous avons calculé *n_x_images* le nombre d'images dont le nom contient un mot du dictionnaire.

L'analyse de différentes balises nous a permis également de calculer d'autres critères tels que :

- n_x_url : nombre de mots du dictionnaire dans l'URL ;
- n_x_meta: nombre de mots du dictionnaire dans la balise meta des mots clés.

5.4.3 Synthèse du vecteur de caractéristiques

Pour récapituler ce qui précède, le vecteur de caractéristique que nous avons proposé d'utiliser pour classifier les pages Web contient les attributs suivants :

- n_mots : nombre total de mots dans la page ;
- n_x_words : nombre de mots de la page figurant dans le dictionnaire ;
- n_images : nombre total d'images ;
- n_x_images : nombre d'images dont le nom contient un mot du dictionnaire ;
- n_liens : nombre total des liens ;
- n_x_liens : nombre de liens contenant des mots interdits du dictionnaire ;
- n_xxx_liens : le nombre de liens répertoriés comme pornographiques dans la liste noire ;
- n_x_url : nombre de mots du dictionnaire dans l'url ;
- n_meta : nombre de mots total dans la balise meta keywords ;
- n_x_meta : nombre de mots du dictionnaire dans la balise meta keywords ;
- pcxwords : pourcentage de mots de la page figurant dans le dictionnaire ;
- pcxmeta : pourcentage de mots-clefs dans les balises méta se trouvant dans le dictionnaire
- pcxliens : pourcentage de liens dont le nom comporte des mots du dictionnaire ;
- pcximage : pourcentage d'images dont le nom comporte des mots du dictionnaire ;

L'extraction des différentes caractéristiques précédentes nécessite donc l'analyse du code HTML d'une page Web. Il nous a donc fallu nous doter de : (1) un client HTTP, qui prend en paramètre une URL et renvoie une page de code HTML ; (2) un analyseur syntaxique (parser HTML), qui lit le code de la page, calcule les valeurs associées aux différents critères et stocke ces valeurs dans une base de données.

Notre parser HTML est fondé sur le principe d'identification séquentielle et linéaire des éléments clefs du langage. En effet, il parcoure le document caractère par caractère, en détectant les débuts et les fins des balises.

Les balises qui nous intéressent dans notre étude sont les suivantes :

- Meta : si l'attribut *name* est « keywords », on recherche des mots du dictionnaire dans l'attribut *content*. Leur nombre est stocké dans *n_x_meta*.
- A : la valeur de l'attribut *href* est traitée – recherche si l'URI cible figure dans les documents déjà traités et marqués « indésirables ». Si c'est le cas, *n_x_liens* est incrémenté.
- Frame : Un objet de classe *Frame*, dérivée de *Site* est créé avec l'URI trouvé dans l'attribut *src*. L'analyse est effectuée pour cette nouvelle URI et les résultats sont additionnés à la page en cours.
- Img : *n_images* est incrémenté - L'attribut *src* est parcouru à la recherche de mots

du dictionnaire, *n_x_images* est incrémenté si un mot est trouvé.

Les mots, une fois délimités, sont comptés (*n_mots*) et soumis à une recherche dans le dictionnaire (*n_x_mots* incrémenté si trouvé).

La figure 5.5 donne un exemple schématisant l'analyse de notre Parser. En jaune, les mots sont du dictionnaire; en violet les liens; en vert, les images. Il est clair que beaucoup de liens ou de noms d'images comportent des mots du dictionnaire sur notre exemple.

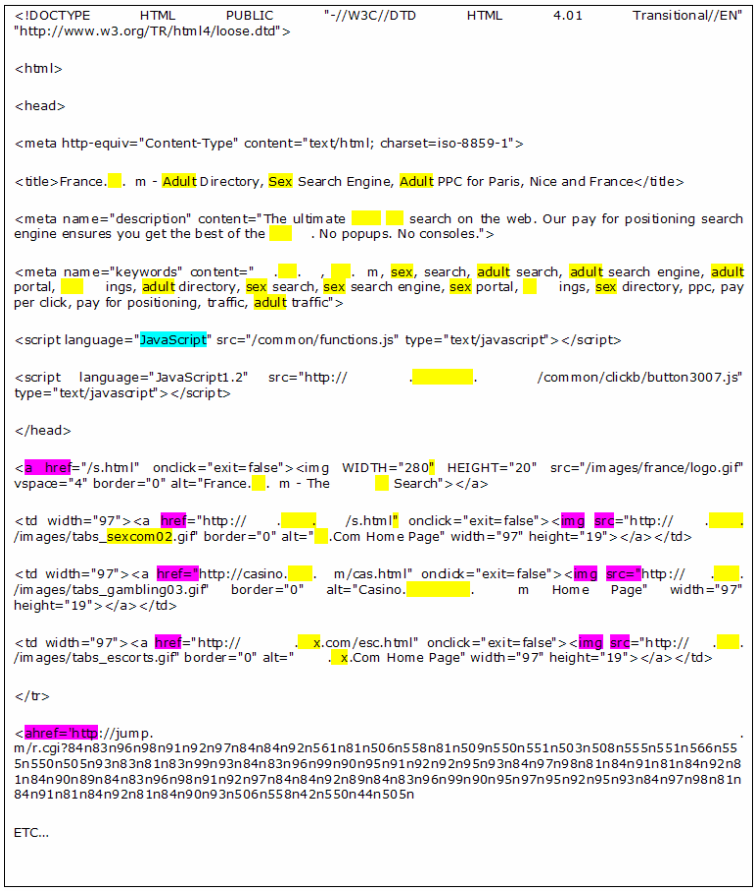


Figure 5. 5. *Exemple d'analyse d'une source HTML*

Sachant que les méthodes de traitement de données exigent souvent une présentation particulière pour le fichier de données, nous avons stocké nos données dans un tableau sous la forme suivante :

Tableau 5. 1. *Extrait de fichier d'apprentissage*

N° site	n_mots	n_x_mots	n_images	n_x_image	n_x_liens	n_x_meta	…	Classe
1	2469	0	24	0	0	0	…	1
…	…	…	…	…	…	…	…	…
1308	526	11	3	0	0	11	…	0
…	…	…	…	…	…	…	…	…

Les lignes de ce tableau représentent les exemples ou les cas à traiter. Ces exemples sont décrits par des attributs et des décisions, qui apparaissent en colonne à l'intersection des lignes et des colonnes, on trouve la valeur de l'attribut en colonne pour l'URL en ligne. La classe de l'URL apparaît dans la dernière colonne et elle a la valeur 0 si le site est adulte sinon la valeur 1.

5.5 Analyse du contenu visuel

Il est un fait que le Web devient de plus en plus visuel et multimédia. Une étude sur plus que 4 millions de pages Web a montré que 70% d'entre elles contiennent des images et qu'il y a en moyenne 18,8 % par page [198]. Ainsi, une classification précise d'un site Web devrait tenir compte de son contenu visuel. Pour notre application particulière, il est une intuition évidente que les sites Web adultes sont caractérisés par un pourcentage élevé de pixels de peau humaine. Par ailleurs, les expérimentations réalisées sur WebGuard utilisant uniquement des caractéristiques issues d'une analyse du contenu textuel et structurel montrent que les sites adultes mal classés sont des sites qui ne contiennent que d'images adultes ou bien des sites où le texte est incorporé dans l'image. Aussi, nous avons décidé de combiner l'analyse du contenu visuel avec celle du contenu textuel et structurel pour affiner les précisions de classification par notre système. S'il est une évidence intuitive que la couleur de peau est un indice important de sites adultes de par ses images pornographiques, il existe cependant plusieurs stratégies d'intégration de cette analyse du contenu visuel pour la classification et le filtrage de ces sites.

Dans la suite de cette section, nous analysons d'abord les différentes stratégies possibles pour intégrer l'analyse du contenu visuel dans la classification de sites. En seconde partie, nous présentons un pré-traitement nécessaire qui nécessite de discriminer les images logo, nombreuses sur Internet, de celles non logo. Une telle discrimination aide à améliorer l'efficacité de classification et de filtrage dans l'intégration du contenu visuel.

5.5.1 Stratégies d'intégration de l'analyseur d'image

Il existe deux stratégies possibles pour intégrer notre analyse du contenu visuel basé sur notre modèle de peau : la première est une stratégie d'homogénéité et elle consiste à ajouter les critères sur le contenu visuel au côté des critères textuels et structurels ; Nous faisons ensuite un nouvel apprentissage pour extraire des nouvelles règles de décision permettant de classer les sites. La deuxième stratégie consiste à appliquer le filtrage de sites basé sur le contenu visuel sur les sites classés sains par notre solution de filtrage basé uniquement sur le contenu textuel et structurel. Si on désigne par WebGuard-TS notre solution de classification et de filtrage par une analyse du contenu textuel et structurel, et par WebGuard-V notre solution de classification et de filtrage basé purement sur le contenu visuel, la deuxième stratégie est donc une stratégie de cascade qui consiste à applique WebGuard-V après WebGuard-TS, le but ici étant de filtrer par WebGuard-V les sites adultes qui ont échappé à la vigilance de WebGuard-TS. Cette deuxième stratégie de cascade a encore deux variantes : la première variante consiste à considérer le nombre d'images classifiées comme image potentiellement pornographique selon une importance de présence de peau et pour cette raison nous désignerons par la suite cette variante par stratégie cascade-%images pornographiques ; La deuxième variante considère l'importance de pixels de peau dans l'ensemble d'images

présentes dans une page Web et nous désignons en conséquence cette variante par stratégie cascade -%peau.

Dans la suite, nous décrivons plus en détail ces trois stratégies. L'évaluation et la comparaison de ces trois stratégies seront développées dans notre section sur les expérimentations.

5.5.1.1 *1ère stratégie d'homogénéité : le contenu visuel utilisé comme d'autres critères sur le contenu textuel et structurel*

Pour cette première stratégie, nous proposons donc d'utiliser, en plus de 14 critères issus de l'analyse du contenu textuel et structurel définis dans la section 5.4.3, les onze critères suivants liés au contenu visuel d'une page Web :

- nombre d'images adultes dans la page Web ;
- pourcentage d'images adultes dans la page Web ;
- nombre d'images adultes dont le nom contient un mot du dictionnaire ;
- pourcentage d'images adultes dont le nom contient un mot du dictionnaire ;
- nombre de logos dans la page Web ;
- pourcentage de logos dans la page Web ;
- nombre de logos dont le nom contient un mot du dictionnaire ;
- pourcentage de pixels de non peau dans la page ;
- nombre d'images saines ;
- pourcentage d'images saines ;
- pourcentage de pixels de peau dans la page Web.

5.5.1.2 *2ème stratégie de cascade : 1ère variante utilisant le pourcentage d'images classées pornographiques (Stratégie de cascade - %ImagesPornographiques)*

L'idée ici est de classifier une page Web selon le pourcentage des images considérées adultes dans celle ci. Ceci dit, il faut déterminer d'abord le pourcentage de pixels de peau à partir duquel une image est classée comme adulte. Ce seuil a été établi par l'étude de 6000 images. Les résultats donnent des moyennes de 18 % pour une base de données d'image non adulte de 4000 fichiers, dont 700 contenants des portraits, et pour les 2000 images adultes, on obtient un taux moyen de 45% de pixels de peau par image. Vu l'écart peu conséquent entre les 2 résultats, nous avons fait le choix de 26% comme seuil limite, afin que plus de 78% des images adultes soient classées comme telles, et que seulement 23 % des images saines dépassent ce critère.

5.5.1.3 *2ème stratégie de cascade : deuxième variante utilisant le pourcentage total de pixels de peau dans une page (Stratégie de cascade - %peau)*

Cette variante nécessite que l'on ait le pourcentage moyen de pixel de peau dans une page à caractère adulte. En utilisant les images contenues dans notre corpus d'apprentissage MYL, les taux révélés par l'incorporation de ce critère donne des résultats du même type que précédemment, l'analyse conduite à une séparation pour un seuil de 24 % (voir figure 5.6). Cependant avec un seuil aussi faible, de nombreux sites non pornographiques sont au-delà du seuil et inversement. Ceci provient de la présence de nombreuses images contenant du texte :

les logos. Après analyse et séparation des images logos, le taux a augmenté à 34%, seuil qui permet de mieux différencier les sites.

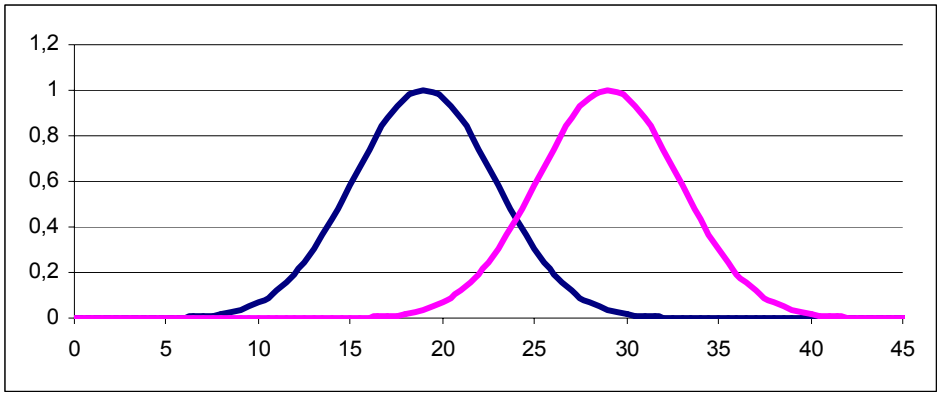
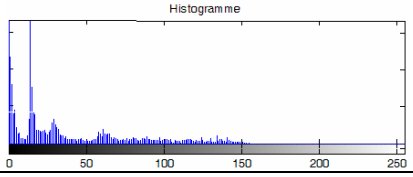
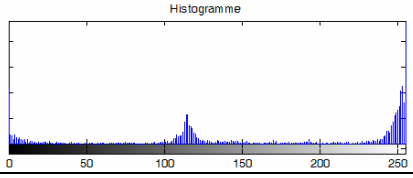
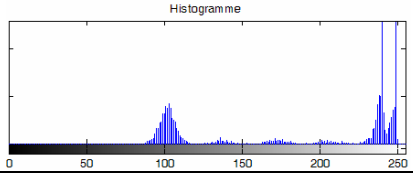


Figure 5. 6. *Gaussiennes des pourcentages de peau pour les sites sain à gauche et adulte à droite*

5.5.2 Identification des images logos

L'analyse des trois stratégies précédentes pour l'intégration de l'analyse du contenu visuel montre qu'un pré-traitement est nécessaire sur les images présentes dans une page Web. En effet, il existe dans une page Web de nombreuses images de logo qui conduisent à fausser le résultat de classification si l'on applique directement sur toutes les images d'une page Web notre analyse du contenu visuel basé sur l'importance de peau dans une image. Il faut appliquer une telle analyse uniquement sur les images non logo. Aussi, nous avons réalisé un pré-traitement qui consiste à discriminer les images logo de celles de non logo. Pour cela, nous avons utilisé une technique simple et efficace. Il s'agit de calculer pour chaque image son histogramme en niveaux de gris et compter le nombre de pics au-dessus d'un seuil que nous avons défini après une étude empirique. La figure 5.7 montre clairement la différence entre les histogrammes issus des images logos et ceux des images non-logos.

Image	*Image niveau de gris*	*Histogramme niveau de gris*	*Classe*
			Image logo
			
			

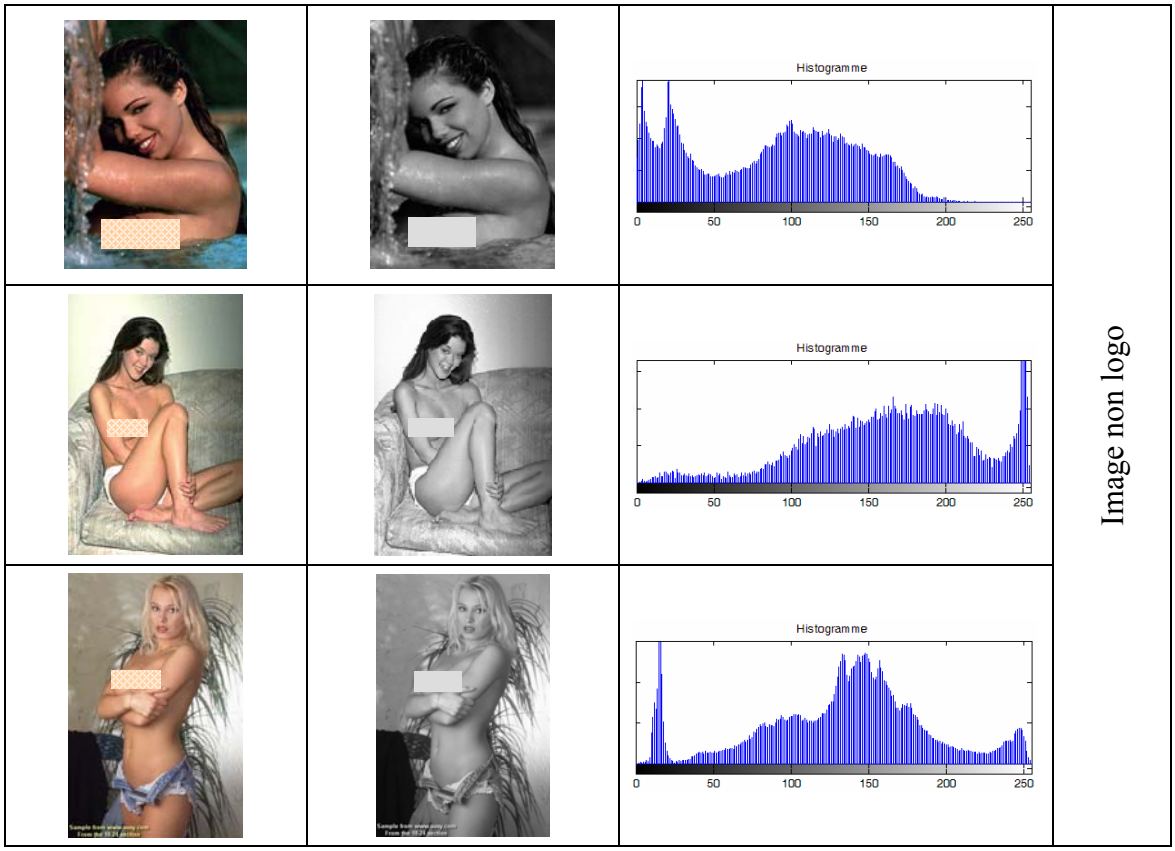


Figure 5. 7. *Histogrammes du niveau de gris des images logo et ceux non-logo*

5.6 Expérimentations et recherche du modèle de prédiction

Cette section présente les différentes expérimentations réalisées afin de trouver le modèle de prédiction le plus pertinent pour notre application. L'étape de recherche du modèle consiste à extraire la connaissance utile de l'ensemble de données que nous avons collecté dans les phases décrites auparavant.

Nous rappelons que dans une démarche d'extraction des connaissances, il y a trois étapes que l'on peut résumer comme il suit :

1. La première porte sur l'élaboration du modèle. Elle fait appel à un échantillon d'apprentissage noté Ω_a dont tous les individus sont décrits dans un espace de représentation noté $\Re$ et appartiennent à l'une des m classes notées. Il s'agit alors, de construire l'application φ qui permet de calculer la classe à partir de la représentation.
2. La seconde étape est celle de la validation. Il s'agit de vérifier, sur un échantillon test Ω_t (n'ayant pas servi à l'apprentissage) dont nous connaissons, pour chacun de ses individus, la représentation et la classe, si le modèle de prédiction φ issue de l'étape précédente donne bien la classe attendue.

3. Enfin, la dernière phase est la généralisation qui consiste à étendre l'application du modèle à tous les individus de la population Ω

Avant de présenter les séries d'expérimentations et afin de clarifier leurs conditions, nous allons décrire brièvement la base d'apprentissage utilisée, ainsi que les conditions et les techniques de validation utilisées.

5.6.1 Base d'apprentissage MYL

La constitution de la base d'apprentissage est une étape importante dans un processus de data mining. En effet, la mauvaise qualité des données complexifie l'apprentissage et nuit à la performance du modèle. Il faut éviter le piège GIGO (Gabage In, Garbage Out), dans lequel les erreurs en entrée entraînent des erreurs en sortie. Cette phase de collecte et de sélection des sites Web constitue une charge de travail considérable.

Pour que l'apprentissage soit efficace, il faut que notre base d'apprentissage soit représentative de la population et que le nombre de sites sur lequel il est fait soit important. Pour cela, nous avons établi une base de données de 2000 sites : 1000 sites adultes et 1000 sites non adultes. Dans les sites adultes on trouve des sites allant de l'érotique au pornographique, dans les sites non adultes on trouve des sites de santé, des sites de lutte contre la pornographie ou le sida etc.

5.6.2 Conditions d'expérimentations et techniques de validation

Dans nos expérimentations, nous présentons deux séries de tests. La première est le résultat de notre système WebGuard-TS, basée uniquement sur une analyse du contenu textuel et structurel des pages Web. La deuxième présente les résultats obtenus après une analyse du contenu textuel et structurel, enrichie par une analyse du contenu visuel. Ceci donne naissance à la version finale de WebGuard appelée par la suite WebGuard-TSV.

Dans la première série d'expérimentations, les variables issues d'une analyse du contenu textuel et structurel des pages Web ont montré leur pertinence : nous avons en effet obtenu un taux de bonne classification élevé. Après une première phase d'apprentissage, sur notre base d'apprentissage, nous avons évalué intensivement la qualité et la stabilité des modèles obtenus à partir des cinq algorithmes de data mining par le biais de trois méthodes : la méthode des taux d'erreur, la méthode de validation croisée et la méthode Bootstrap.

Les mesures d'évaluation que nous avons effectuées sont celles définies dans la section 8 du chapitre 3, à savoir le taux d'erreur global, le taux d'erreur a priori et le taux d'erreur a posteriori. Rappelons que le taux d'erreur global est le complément du taux de classification tandis que le taux d'erreur a priori (respectivement le taux d'erreur a posteriori) est le complément du taux de rappel classique (respectivement le taux de précision). Ainsi, plus le taux d'erreur a priori obtenu est faible, meilleur est le taux de rappel. La même règle s'applique au rapport entre le taux d'erreur global et le taux global de classification ainsi qu'entre le taux d'erreur a posteriori et le taux de précision. Dans la suite nous détaillons les principes des différentes méthodes d'évaluation ainsi que les conditions d'expérimentations.

La **méthode des taux d'erreur** consiste à séparer la base d'apprentissage en deux sous-ensembles : un pour l'apprentissage et l'autre pour le test. Dans notre cas, nous avons réservé 70% des sites Web pour l'apprentissage et le reste, c'est à dire les 30%, a été utilisé pour le test. La base d'apprentissage sert donc à construire des modèles à partir des cinq algorithmes de data mining et la base de test sert de son coté à vérifier la stabilité de ces modèles. Ce processus est répété trois fois avec des sous-ensembles choisis à chaque fois aléatoirement. Pour calculer les taux d'erreur finaux nous faisons la moyenne des différents taux d'erreur calculés au bout de chaque test. Pour des raisons de simplicité, nous avons utilisé, durant nos expérimentations, la méthode des taux d'erreur en tant que méthode principale d'évaluation. Lorsqu'un algorithme de data mining produit des résultats acceptables selon cette technique d'évaluation, nous validons davantage ces résultats avec les deux autres techniques, la cross validation et le Bootstrap.

La **validation croisée** propose de diviser la base d'échantillons en "*s*" parties égales, avec apprentissage sur les (*s*-1) de la base, et un test sur la partie restante. Ensuite, on effectue une permutation des bases testées, ce qui donne naissance à un tableau de confusion. Ce dernier présente la moyenne des "*s*" tests effectués ; dans nos expérimentations s prend la valeur 10.

Le **Bootsrap** est une technique empirique qui permet d'estimer l'espérance mathématique du biais d'optimisme de l'estimation par resubstitution. Le principe est fondé sur *B* jeux d'apprentissage constitué par un tirage aléatoire simple avec remise dans la base d'apprentissage. Il est souvent recommandé de procéder à au moins une centaine de répétitions pour espérer avoir une bonne fiabilité. En pratique, nous avons effectué 100 répétitions et nous avons ajusté les paramètres de telle sorte que chaque site de notre base ait une probabilité de 36,8% pour figurer dans le sous-ensemble d'apprentissage.

Cependant, les algorithmes utilisés ont parfois des difficultés à distinguer les règles liées à l'échantillon (qui n'ont aucune valeur) de celles qui peuvent être généralisées, et il est fréquent que certains classifieurs « apprennent » les données plutôt que le modèle. Par exemple, si dans le fichier d'apprentissage, tous les sites qui ont des images sont des sites adultes, le système en conclura que tous sites ayant des images sont des sites adultes. Afin d'éviter ce phénomène de « surapprentissage » (overfitting), nous avons testé nos modèles sur la base de test MYL composé de 200 sites adulte et de 200 sites non adulte comme c'est décrit dans la section 5.2.1. Un modèle performant donnera normalement des résultats proches sur la base d'apprentissage et sur la base de test.

Dans la deuxième série d'expérimentations nous avons intégré notre technique d'analyse du contenu visuel en vue d'améliorer les performances de notre système de filtrage.
Dans ce qui suit, nous décrivons plus en détails ces deux séries d'expérimentations.

5.6.3 Résultats basés seulement sur une analyse du contenu textuel et structurel

Ici, seulement les variables extraites des contenus textuels et structurels ont été utilisées. Nous avons étudié cinq algorithmes de data mining qui sont : ID3, C4.5, Improved C4.5, Sipina avec λ=5,22 et une contrainte d'admissibilité fixée à 20 noté Sipina (5.22) et Sipina avec λ=12 et une contrainte d'admissibilité de 50 notée Sipina (12).

5.6.3.1 *Méthode des taux d'erreur*

Dans cette série de tests nous avons évalué les résultats obtenus par les différents algorithmes au moyen des trois mesures d'évaluation, le taux d'erreur a priori, le taux d'erreur a posteriori et le taux d'erreur globale afin de chercher les classifieurs les mieux adaptés. La figure 5.8 montre les résultats obtenus, pour chaque algorithme et son taux d'erreur.

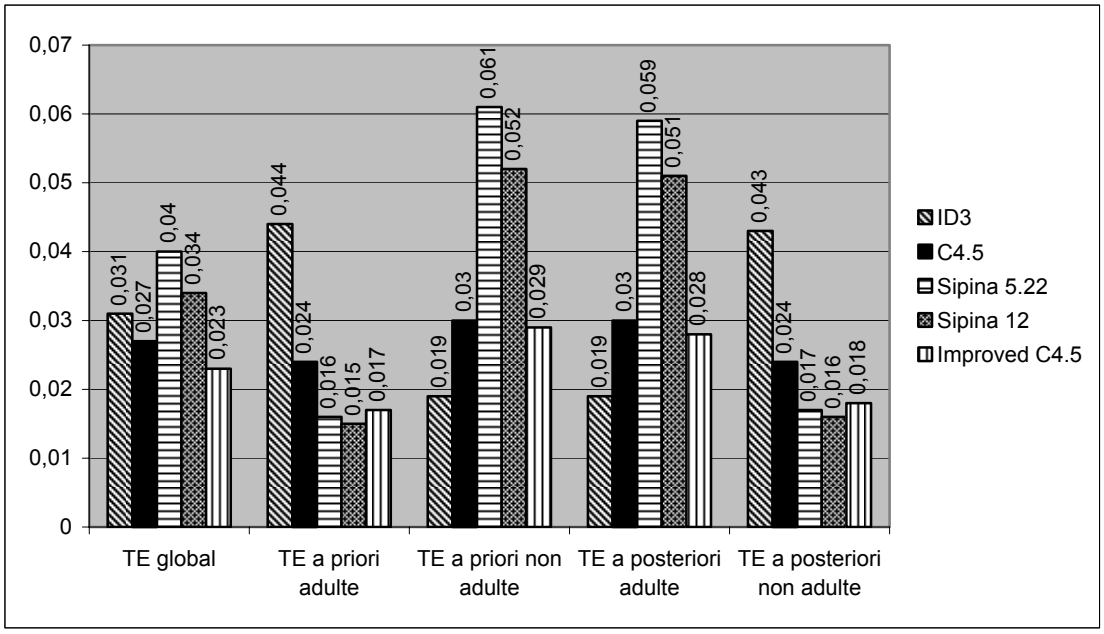


Figure 5. 8. *Résultats des évaluations par la méthode des taux d'erreur*

D'après ces résultats, le taux d'erreur global, pour les cinq algorithmes, est quasiment le même. Ce taux ne dépasse pas 4%. On remarque aussi qu'il y a clairement une complémentarité entre le taux d'erreur a priori adulte et celui non adulte, tout comme entre le taux d'erreur a posteriori adulte et celui non adulte. Par exemple, Sipina (12) donne un meilleur taux de classification des sites adultes avec un taux d'erreur a priori de 1,5% mais enregistre par contre un taux d'erreur a priori de 5,2% pour la classe non adulte. On a également observé le même phénomène du coté du taux d'erreur a posteriori. Sipina (12) a réalisé le meilleur résultat de classification des sites non adultes avec un taux d'erreur a posteriori de 1,7%, en revanche il a montré un mauvais taux d'erreur a posteriori pour la classe adulte avec 5.9%.

Il est à noter que sur cette série de tests l'algorithme Improved C4.5 donne les meilleurs résultats, avec un taux d'erreur global de 2,3% et des taux d'erreur a priori et a posteriori qui s'étendent de 1,7% à 2,9% sur la classification des sites adultes et des sites non adultes.

5.6.3.2 *Validation croisée et Bootstrap*

Les résultats expérimentaux obtenus par la technique des taux d'erreur sont donc très encourageants. En effet, les cinq classifieurs, basés uniquement sur des variables textuelles et

structurelles donnent des résultats qui surpassent de 6% la meilleure performance réalisée par les produits commerciaux.

Pour valider ces résultats, nous avons utilisé les deux autres techniques d'évaluation. Les figures 5.9 et 5.10 présentent respectivement les résultats expérimentaux de la validation croisée et du Bootstrap.

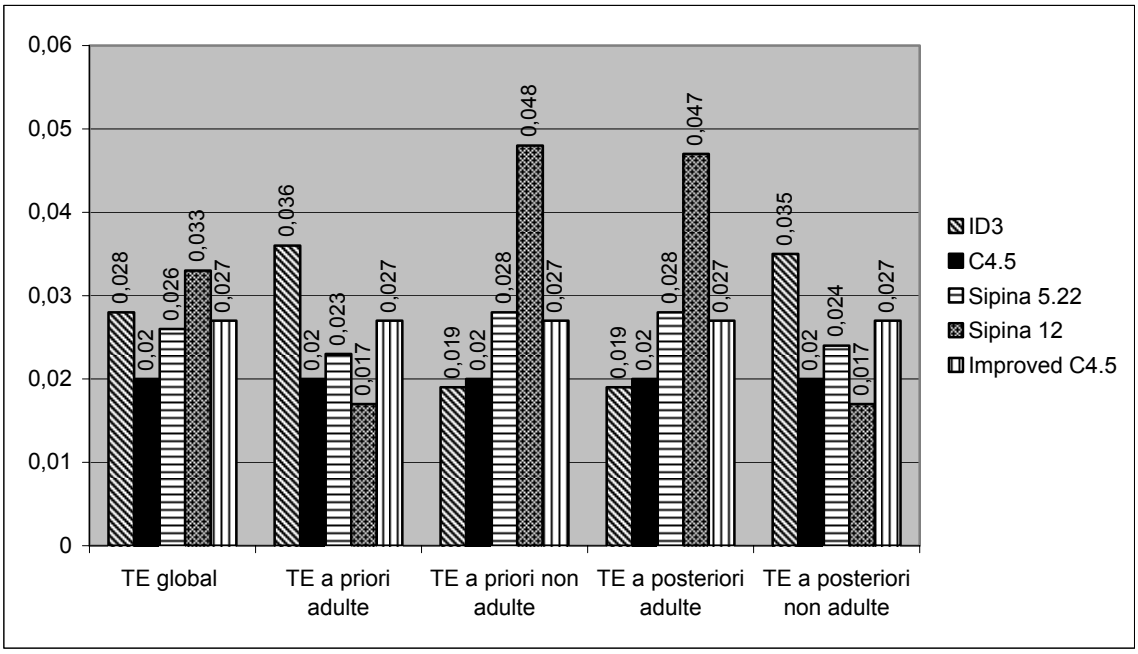


Figure 5. 9. *Résultats des évaluations par la méthode de validation croisée*

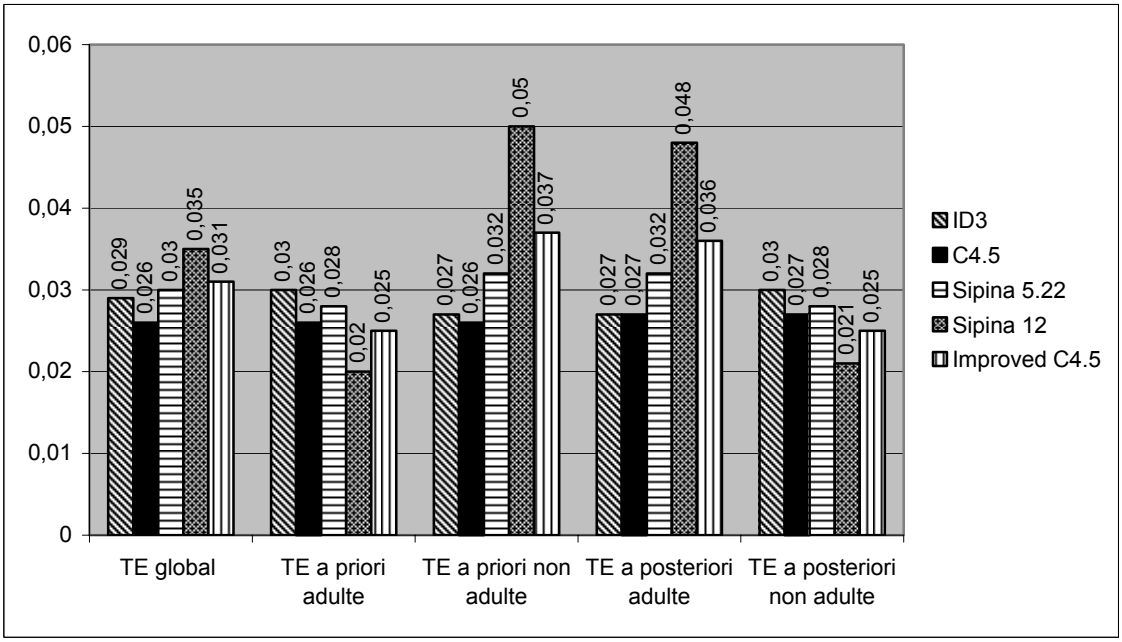


Figure 5. 10. *Résultats des évaluations par la méthode Bootstrap*

Globalement, les résultats obtenus pour chaque algorithme sont semblables. Néanmoins, nous pouvons signaler que pour ID3, C4.5 et Sipina (5,22) les taux d'erreurs globaux obtenus par la méthode des taux d'erreur sont plus élevés que ceux relevés avec les deux autres méthodes. Cela signifie que les taux obtenus par la méthode des taux d'erreur sont approximativement le maximum de ceux obtenus par la méthode de validation croisée ou par la méthode Bootstrap. Par conséquent, nous pouvons conclure que ces taux sont faibles (moins de 4%).

Concernant les deux algorithmes Improved C4.5 et Sipina (12), ils présentent des taux d'erreur globaux moins élevés avec la méthode des taux d'erreur. Mais ces taux ne diffèrent que de 1% par rapport aux taux obtenus par les deux autres méthodes d'évaluation.

La figure 5.11 résume les résultats d'évaluation de chaque classifieur, produit par chaque algorithme de data mining, selon les trois méthodes d'évaluation.

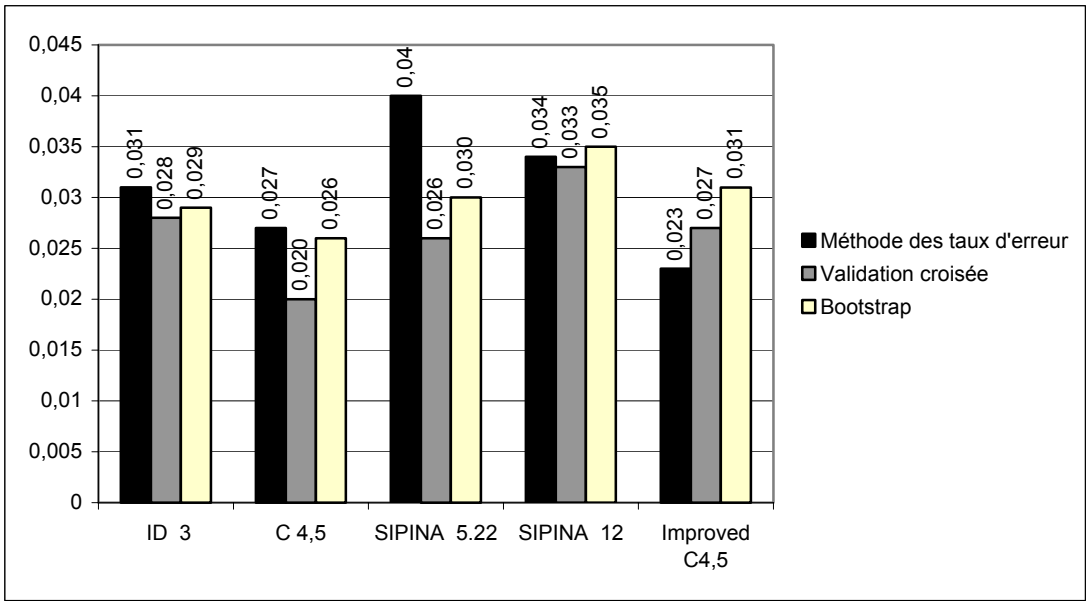


Figure 5. 11. *Comparaison globale des résultats des trois techniques d'évaluation*

Pour conclure, nous pouvons dire que le taux d'erreur global de chaque algorithme est inférieur de 4 %. Il reste maintenant à vérifier les modèles obtenus sur notre base de test MYL afin de s'assurer de la stabilité et la pertinence des résultats.

5.6.3.3 *Résultats expérimentaux sur la base de test MYL*

Encouragés par les résultats précédents, nous avons alors testé les cinq algorithmes de data mining sur notre base de test MYL. Nous rappelons que cette base a été employée pour comparer les produits commerciaux dans la section 5.2.3. Les résultats expérimentaux sont récapitulés par la figure 5.12.

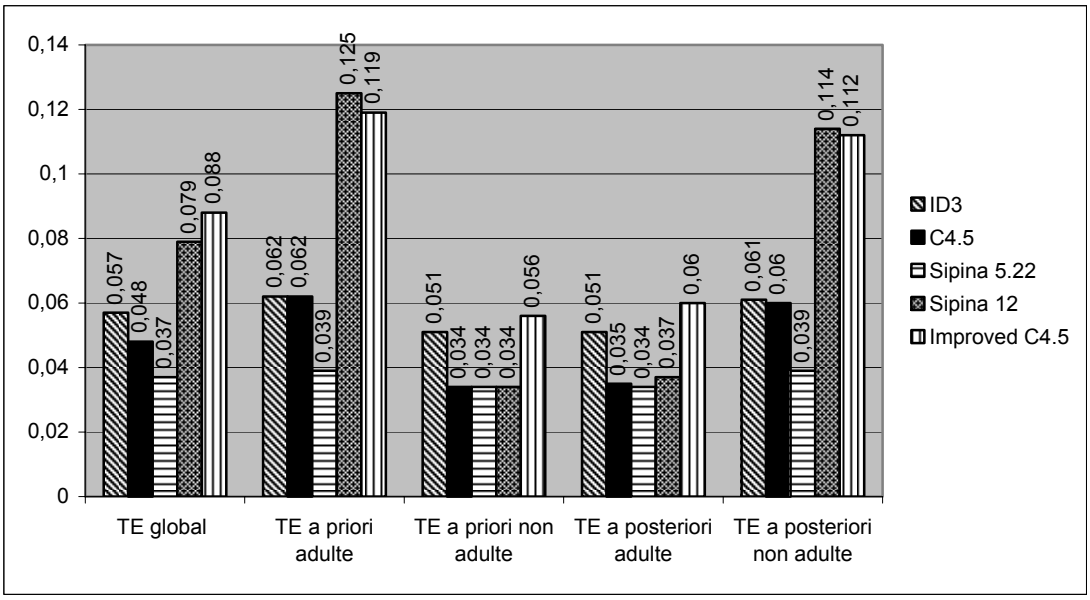


Figure 5. 12. *Résultats expérimentaux sur la base de test MYL par les cinq algorithmes*

Comme nous pouvons voir dans cette figure, le taux d'erreur global moyen pour l'ensemble des cinq algorithmes de data mining est d'environ 6%, ce qui correspond à un taux de classification de 94%, soit 4% de mieux que la meilleure performance des produits commerciaux évalués.

Dans notre stratégie de classification des sites en adulte/non-adulte, nous préférons que quelques sites à caractère « moyen » si l'on peut dire, comme des sites de sexologie, soient filtrés plutôt que d'avoir des sites adultes non filtrés. Il est plus important de bloquer un site adulte que de laisser passé un site non adulte. Pour cette raison, nous avons utilisé le taux d'erreur a priori adulte pour la prise de décision. Ce taux d'erreur est le plus important pour l'évaluation parce qu'il mesure l'efficacité de classifier des sites adultes.

Les taux d'erreur a priori adultes montrent que Sipina (12) et Improved C4.5 ne sont pas aussi efficace que ID3, C4.5 et Sipina (5,22). En plus le taux d'erreur a priori non adulte donnée par l'algorithme improved C4.5 est élevé. Ce taux prouve que cet algorithme ne fournit pas une décision fiable en ce qui concerne la classification des sites non adultes. Malgré ces remarques nous pouvons constater que les résultats de ces deux algorithmes sont satisfaisants.

Nous observons encore une fois une complémentarité entre le taux d'erreur a priori sur les sites adultes et non adultes de même pour le taux d'erreur a posteriori entre les deux classes. Sipina (12) a montré, ici, la plus mauvaise performance sur la classification de sites adultes, il a réalisé en même temps la meilleure performance du taux d'erreur a priori sur la classification de sites non adultes.

Il est clair après cette série de tests que les cinq algorithmes donnent des résultats encourageants. Ainsi, vu que chacun des algorithmes possède ses propres règles, nous avons initialement opté pour un système électif en ce qui concerne la prise de décision du logiciel. C'est-à-dire que celui-ci utilisait cinq algorithmes de data mining en parallèle pour générer

cinq arbres de décision différents après apprentissage. Ainsi, lors de l'utilisation du logiciel, nous avons obtenu de la part de chaque algorithme une réponse classifiant la page Web traitée. Le logiciel agissait alors selon la décision prise par une majorité d'algorithmes. Par exemple, si trois algorithmes sur cinq déclarent une page adulte, elle sera bloquée. L'avantage de cette méthode semble évident : une décision est plus fiable si elle a été prise par plusieurs algorithmes différents. Cette politique donne naissance à notre premier système de filtrage, WebGuard-TS.

En testant les performances de WebGuard-TS sur la base de test MYL, nous avons obtenu les résultats décrits par la figure 5.13.

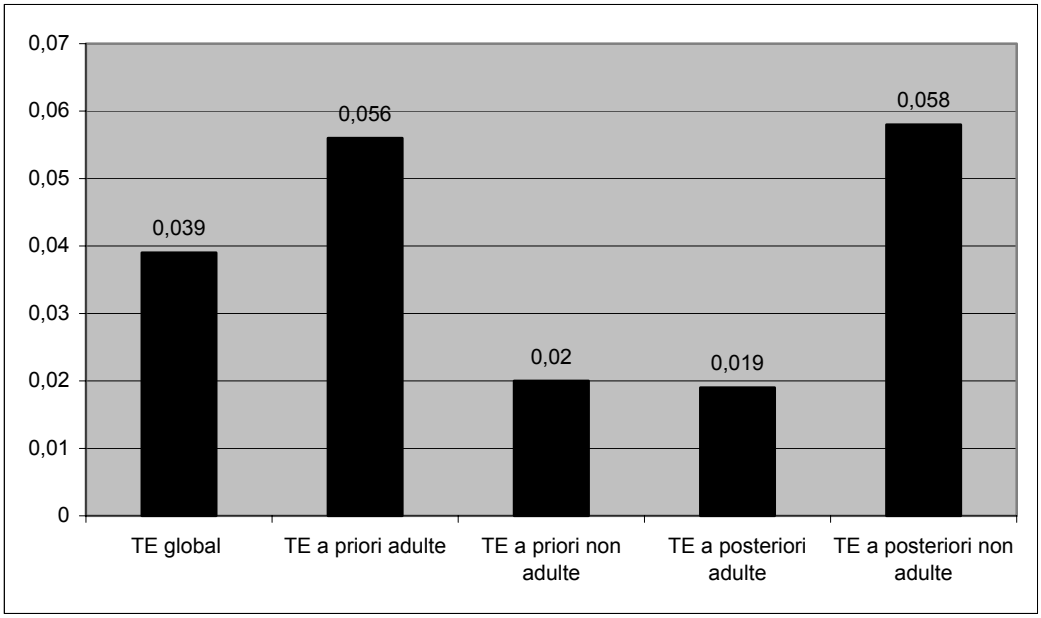


Figure 5. 13. *Résultats expérimentaux par vote majoritaire sur la base de test MYL (WebGuard-TS)*

Comme nous pouvons le constater sur la figure 5.13, WebGuard-TS produit un taux d'erreur global de 3.9% en se basant seulement sur des attributs textuels et structurels ce qui correspond a un taux de bonne classification de 96.1% qui surpasse le meilleur produit testé de 6.1%. Ces résultats montrent aussi l'apport de la combinaison des différentes techniques de data mining pour notre application.

5.6.4 Résultats après intégration de l'analyse du contenu visuel

Bien que WebGuard-TS montre un taux d'erreur global faible, sa performance peut encore être améliorée par l'analyse du contenu visuel en se basant sur notre technique de détection des régions de peau dans l'image. Dans la section 5.5.1, nous avons présenté deux stratégies d'intégration possibles de l'analyse du contenu visuel basé sur la couleur de peau : une première stratégie dite d'homogénéité qui consiste à utiliser des critères supplémentaires liés au contenu visuel à côté des critères issus d'une analyse du contenu textuel et structurel d'une page Web, puis une deuxième stratégie dite de cascade qui consiste à appliquer WebGuard-V basé sur l'analyse du contenu visuel uniquement, sur les résultats obtenus par WebGuard-TS.

Rappelons que cette deuxième stratégie de cascade peut aussi avoir deux variantes possibles : variante Cascade - %images pornographiques qui compte dans une page Web le pourcentage d'images potentiellement pornographiques au regard de l'importance de peau, puis la variante cascade -%peau qui considère l'importance de pixels de peau dans l'ensemble d'images présentes dans une page Web.

5.6.4.1 *1ère stratégie d'homogénéité : le contenu visuel utilisé comme d'autres critères sur le contenu textuel et structurel*

Pour cette première stratégie, rappelons que nous proposons donc d'utiliser, en plus de 14 critères issus de l'analyse du contenu textuel et structurel définis dans la section 5.4.3, les onze critères suivants liés au contenu visuel d'une page Web :

- nombre d'images adultes dans la page Web ;
- pourcentage d'images adultes dans la page Web ;
- nombre d'images adultes dont le nom contient un mot du dictionnaire ;
- pourcentage d'images adultes dont le nom contient un mot du dictionnaire ;
- nombre de logos dans la page Web ;
- pourcentage de logos dans la page Web ;
- nombre de logos dont le nom contient un mot du dictionnaire ;
- pourcentage de pixels de non peau dans la page ;
- nombre d'images saines ;
- pourcentage d'images saines ;
- pourcentage de pixels de peau dans la page Web.

La figure 5.14 résume les résultats obtenus sur notre base de test selon les cinq algorithmes.

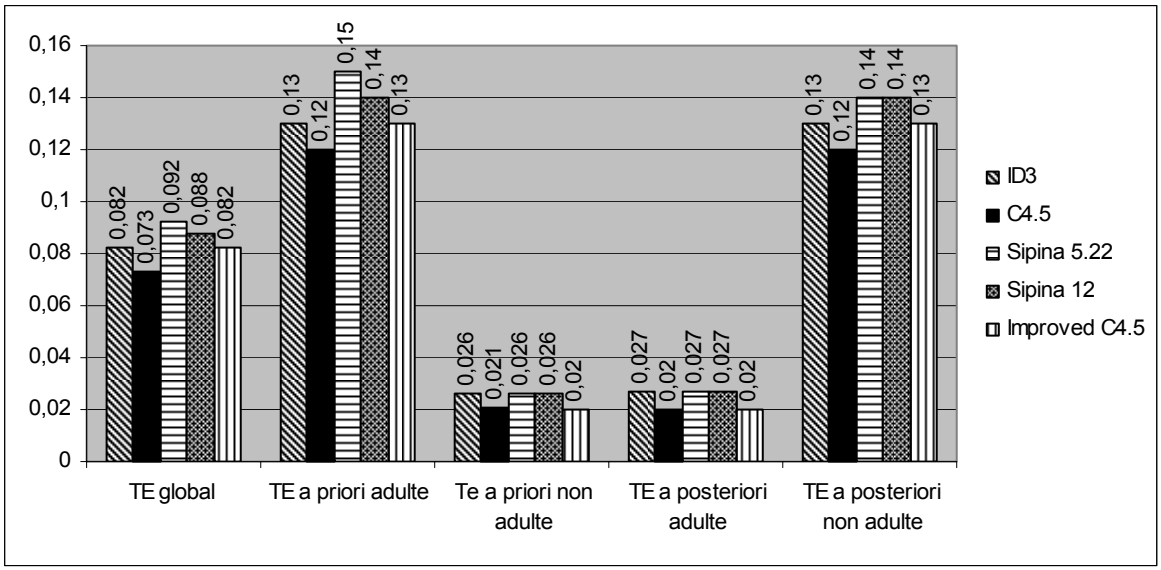


Figure 5. 14. *Résultats expérimentaux de la 1ère stratégie*

On peut remarquer de cette figure que les résultats sont moins bons que WebGuard-TS qui utilise uniquement l'analyse du contenu textuel et structurel. Non seulement les critères liés

au contenu visuel n'ont pas permis une meilleure différenciation selon cette stratégie, en plus ils ont introduit un bruit lors de l'apprentissage.

5.6.4.2 *2ème stratégie de cascade - variante pourcentage d'images classées potentiellement pornographiques*

L'idée ici est de classifier une page Web selon le pourcentage des images considérées adultes dans celle ci. Rappelons que nous avons fixé un seuil de 26% pour discriminer les sites normaux des sites pornographiques à partir d'une étude de 6000 images. En incorporant le pourcentage d'images adultes dans le reste des critères on obtient les résultats de la figure 5.15.

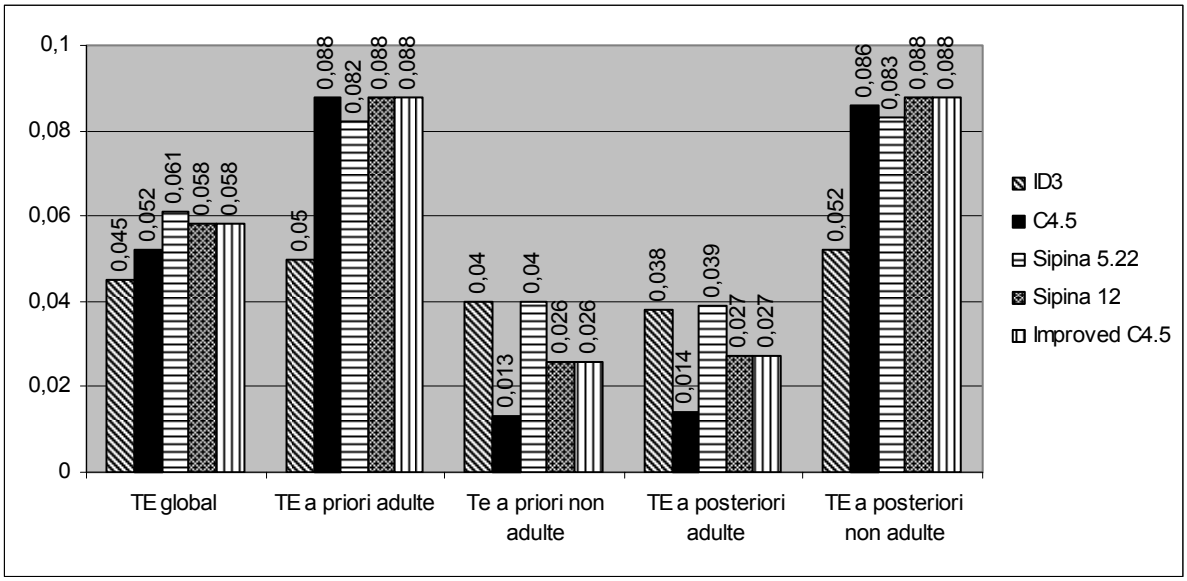


Figure 5. 15. *Résultats expérimentaux de la 2ème stratégie – variante ImagesPornographiques*

On remarque une faible amélioration en utilisant cette stratégie. Ceci vient du fait que le seuil qui discrimine les sites normaux des sites pornographiques est trop faible.

5.6.4.3 *2ème stratégie de cascade : variante pourcentage total de pixels de peau dans une page*

Cette fois-ci, nous proposons d'utiliser le pourcentage total de pixels de peau dans une page Web comme critère de discrimination entre sites normaux et sites pornographiques. Encore une fois, nous avons besoin de déterminer un seuil basé sur ce pourcentage permettant de discriminer les sites normaux des sites adulte. En utilisant notre base d'apprentissage, on aboutit à un seuil de 24% (voir figure 5.16). En utilisant un tel seuil pour la classification des sites préalablement classifiés comme étant sains par WebGuard-TS, nous obtenons des taux de classification très similaires aux résultats de la variante précédente. Cependant, si l'on réalise un pré-traitement en enlevant les images de logo nombreuses dans les pages Web, le seuil sur le pourcentage de peau dans une page a été ramené à 34% qui permet de discriminer beaucoup mieux les sites adulte de ceux normaux.

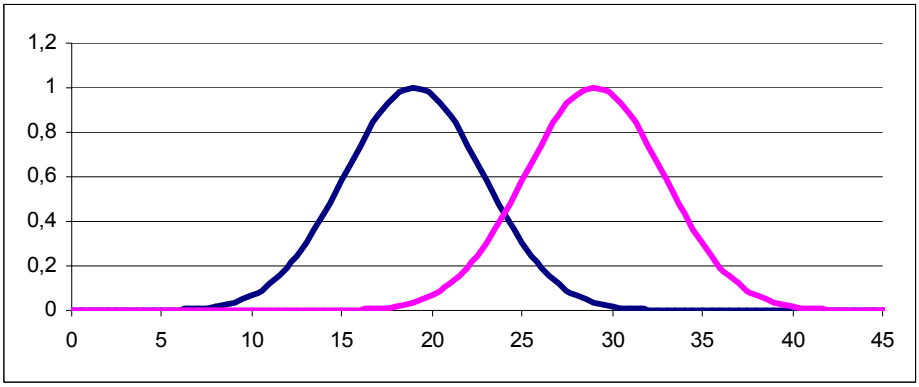


Figure 5. 16. *Gaussiennes des pourcentages de peau pour les sites sain à gauche et adulte à droite*

5.6.4.4 *Synthèse*

L'étude de ces trois modes d'intégration du contenu visuel nous conduit à garder la deuxième stratégie de cascade avec la variante utilisant le pourcentage total de peau dans les images d'une page Web. Rappelons que la solution résultant d'un tel filtrage basé uniquement sur le contenu visuel a été appelée WebGuard-V. Notre solution de classification et de filtrage globale, WebGuard-TSV, est donc un procédé à deux étages cascadés : Dans une première étape, nous utilisons notre système de filtrage WebGuard-TS qui a déjà donné un bon taux de classification. Dans une deuxième étape, pour résoudre le problème des sites mal classés et pour affiner notre technique de filtrage nous appliquons WebGuard-V basé uniquement sur le pourcentage de peau dans une page Web.

Afin de montrer l'apport de l'analyse du contenu visuel pour une telle application de filtrage, nous avons comparé les résultats obtenus par WebGuard-TS et WebGuard-TSV, sur la base de test MYL. La figure 5.17 illustre l'optimisation de notre système.

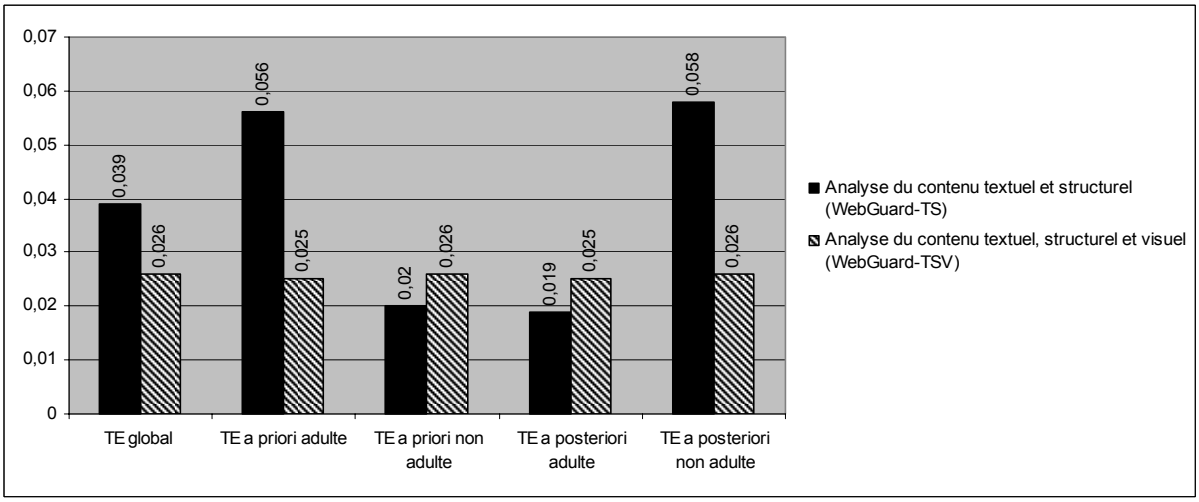


Figure 5. 17. *Comparaison des taux de classification de WebGuard-TS et WebGuard-TSV*

Comme nous pouvons le constater WebGuard-TSV a nettement amélioré la performance obtenue par WebGuard-TS, en réalisant un taux d'erreur a priori adulte de 2.5%, et un taux d'erreur globale de 2.6% seulement.

Nous avons également comparé notre système WebGuard-TSV avec les systèmes de détection et de filtrage de contenu adulte précédemment mentionnés, à savoir Cyber Patrol[197], Norton Internet Security[195], PureSight[196], Cybersitter[193], Net Nanny[194], IE (Internet Explorer)[192]. La figure 5.18 montre la performance de notre système par rapport aux produits déjà présents sur le marché.

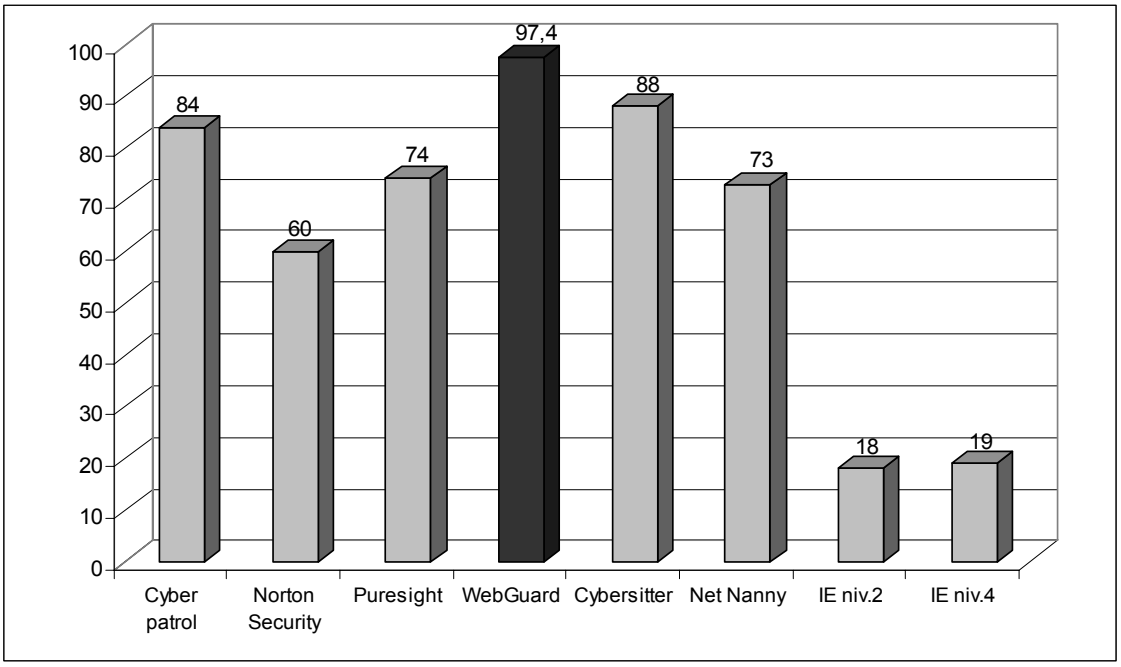


Figure 5. 18. *Comparaison de WebGuard-TSV avec les logiciels existants*

Ces résultats nous ont encouragé à expérimenter plus WebGuard sur une liste noire de 12311 sites adultes manuellement rassemblés et classifiés par le ministère français de l'éducation nationale. Les résultats sont décrits par les tableaux 5.2 et 5.3 qui représentent respectivement les résultats de WebGuard-TS et WeGuard-TSV.

Tableau 5. 2. *Résultats de la classification des sites du ministère de l'éducation nationale par WebGuard-TS*

WebGuard-TS	Résultats de classification	
Types de sites	Sites adultes	Sites sains
Sites adultes	5819	1046
Sites sains		1723
Sites disparus	3723	
Taux de succès : 87.82		

Tableau 5. 3. *Résultats de classification des sites du ministère de l'éducation nationale par WebGuard-TSV*

WebGuard-TSV	Résultats de classification	
Types de sites	Sites adultes	Sites sains
Sites adultes	6489	376
Sites sains		1723
Sites disparus	3723	
Taux de succès : 95.62		

D'après ces tableaux, 1723 sites ont été classés comme sains par notre logiciel et après une vérification manuelle de ces sites nous avons trouvé que ce sont effectivement des sites sains, d'où l'importance d'une analyse dynamique des pages Web, car plusieurs sites changent de contenu d'un jour à un autre. Nous montrons aussi par cette expérimentation l'importance de notre analyseur d'image : ce dernier a amélioré le taux de succès de classification de 87.82 à 95.62%.

5.7 Implémentation

5.7.1 Pondération des différents algorithmes utilisés

Nous avons vu dans la section 5.2.3, qu'un inconvénient relativement important de la plupart des logiciels est le réglage grossier du niveau du filtrage. Il s'agit d'une fonctionnalité qui nous apparaît importante. En effet, étant donnée la diversité des cultures les gens voient les choses différemment, aussi bien pour le caractère adulte ou non adulte des sites.

Nous avons donc répondu à ce besoin en installant un système basé sur la combinaison des cinq algorithmes de sorte que nous puissions changer la sensibilité de filtrage en déplaçant un seuil.

Comme nous avons constaté dans la figure 5.12, les cinq algorithmes de data mining ne sont pas tous aussi efficaces les uns que les autres. Leur donner tous le même poids nous paraît donc un peu trop simple et un système de choix pondéré peut constituer une amélioration non négligeable.

5.7.1.1 *Principe de la pondération*

Nous avons effectué des tests rigoureux et documentés sur les différents algorithmes de data mining. Nous disposions de données précises permettant de faire un choix réfléchi quant aux algorithmes à utiliser pour le filtrage. Concrètement, au niveau de l'implémentation dans le code, nous avons cherché une formule mathématique utilisant les taux d'erreur a priori qui donnerait à l'algorithme un coefficient d'autant plus grand qu'il est efficace. Nous avons choisi d'utiliser le taux d'erreur a priori car il correspond au taux de réussite relatif à chaque classe. Les taux d'erreur utilisés dans notre démarche sont celles présentés par la figure 5.12.

La formule obtenue remplit toutes ces conditions.

$$c_i = \frac{\alpha_i}{\sum_{i=1}^{N} \alpha_i} \quad \text{avec} \quad \alpha_i = (1-(\tau_i - s))^n$$

Où on note :

τ_i : le taux d'erreur a priori de l'algorithme i
N : le nombre d'algorithmes utilisés
n : la puissance utilisée dans le calcul pour accentuer les différences
s : le seuil d'erreur enlevé pour accentuer par la suite les différences en les valeurs.
c_i : le coefficient de pondération utilisé dans le code pour la décision du logiciel

Nous utilisons N=5 c'est à dire cinq algorithmes de data mining : ID3, C4.5, Improved C4.5, Sipina λ = 5,22 et Sipina λ = 12. Comme nous avons obtenu des taux dont le plus faible était juste au-dessus de 0,03, nous avons choisi *s=0,03*. Après plusieurs essais successifs, nous avons déterminé que n = 5 donnait le meilleur résultat. Le tableau 5.4 présente les coefficients attribués aux différents algorithmes.

Tableau 5. 4. *Coefficients des différents algorithmes*

	ID3	C4.5	Improved C4.5	Sipina (12)	Sipina (5,22)
Coefficient	0,25	0,25	0,09	0,08	0,33

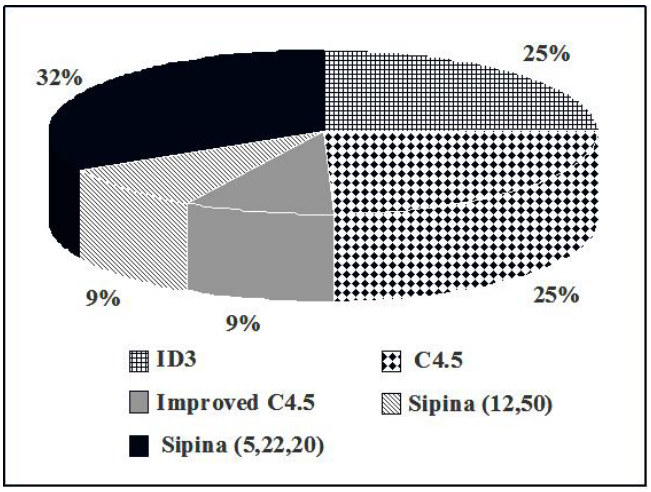


Figure 5. 19. *Pondération des différents algorithmes*

Ensuite, pour la pondération, le logiciel calculera le nombre suivant :

$$\rho = \sum_{i=1}^{n} c_i r_i$$

où r_i est une variable binaire représentant la réponse de l'algorithme i : r_i = 0 si le site est considéré non pornographique et 1 autrement.

Ainsi, on obtient une valeur ρ d'autant plus grande que le site a de chances d'être pornographique et qui donne un coefficient important aux algorithmes. Il est ensuite possible d'implémenter une fonction de sensibilité du filtre en laissant l'utilisateur fixer le seuil ρ_{min} au-dessus duquel le site sera considéré pornographique par notre logiciel.

5.7.1.2 *Apport de la pondération*

Pour montrer l'apport de la pondération nous avons testé notre méthode sur 163 sites adultes en utilisant les 5 algorithmes. Pour chaque algorithme on obtient un verdict de classification, 0 pour un site sain et 1 pour un site adulte. Il y a donc un chiffre pour chaque algorithme utilisé dans le processus de pondération dans l'ordre suivant : ID3, C4.5, Improved C4.5, SIPINA (12), SIPINA (5,22) à partir de ces chiffres et des coefficients de pondérations nous calculons donc la valeur ρ comme décrit dans la section précédente.

Après les tests que nous avons réalisés, nous pouvons tout d'abord dire que les sites sont très majoritairement classés de manière unanime par les algorithmes utilisés. Le désaccord intervient néanmoins sur une proportion non négligeable de sites. C'est là où l'apport de la pondération apparaît. Dans le cas de non-unanimité, 81.8% de ces sites sont classés par 3 algorithmes comme pornographiques, ce qui, avec la pondération, donne un verdict final correct. Cependant, un système électif suivant la décision majoritaire aurait donné le même résultat. Le réel intérêt de la pondération apparaît sur les 18.2% de sites qui ne sont classés comme pornographiques que par 1 ou 2 algorithmes. La pondération donne un chiffre moins élevé que pour les autres cas mais qui peut tout de même être pris en compte par un réglage assez restrictif du filtre. D'où l'amélioration.

Exemple 1 : Sites avec 3 votes pour adulte

Ces sites représentent 81.8% des sites sur lesquels les algorithmes ne sont pas unanimes. La répartition des votes varie. Exemple typique sur ces deux sites présentés par le tableau 5.5.

Tableau 5. 5. *Sites avec 3 votes pour adultes*

Site	Verdicts de classification	ρ
http://www.7emeciel.com/paradis.html	10011	0.5
http://www.madamesalope.com	11100	0.83

Sans pondération, le statut de ces deux sites est le même alors qu'avec, l'un est plus sûrement adulte que l'autre. La classification est donc affinée. Même chose pour les sites suivants :

http ://www.jf18ans.com/photos.html ?id=10000&md=0	11100
http ://www.milfhunter.com/main.htm ?id=	10011
http ://www.oversexe.com/members/index.php ?l=0	11100
http ://www.sexy-beast.net/errors/404.html	11100
http ://www.silkyblondes.com/index2.shtml ?	10011
http ://www.sublimanal.com/index.html ?p=pdv&id=11924&e=0&w=0	11100
http ://www.superpoitrine.com/index.html ?&id=10448	11100

Exemple 2 : Sites avec 1 ou 2 votes pour adulte

Ces sites représentent 9.1% des sites sur lesquels les algorithmes ne sont pas unanimes. Le tableau 5.6 présente 2 exemples, le premier exemple avec 2 votes pour adultes et le deuxième avec juste un vote.

Tableau 5. 6. *Sites avec 1ou 2 votes pour adultes*

Site	Verdicts	ρ
http://www.mega-galerie.com/pages/index2.php3%20?id=1821	10010	0.41
http://hardcore.freepornsexpics.com/ ?f	00100	0.25

Sur ces deux derniers exemples, le réglage plus fin que la pondération permet aurait classé ces sites comme adulte car l'analyse prévoyait qu'ils devaient contenir des éléments non sains. Le système de vote des algorithmes aurait tout simplement classé ces sites comme sains.

La pondération permet donc à l'utilisateur, s'il le désire, de filtrer les sites à caractère douteux, c'est-à-dire pouvant contenir des éléments à caractère adulte avec une certaine probabilité.

Nous signalons que le principe du vote majoritaire utilisé dans les expériences précédentes correspond a $\tau = 0{,}42$.

5.7.2 Présentation de l'interface graphique de WebGuard

Nous présentons dans cette section l'interface graphique de notre logiciel WebGuard et les différentes fonctionnalités. Le grand avantage de cette interface est de permettre une utilisation simple car il est difficile de faire fonctionner un programme en ligne de commande avec tous les arguments nécessaires (adresse URL, méthode de filtrage, etc.). Cela permet également de faire une démonstration du logiciel, plus simplement.

5.7.2.1 *Boite de dialogue principale*

La boîte de dialogue principale de notre logiciel se compose de différents objets :

- une zone de texte dans laquelle est écrit le résultat de l'analyse de la page
- une zone de texte à éditer où l'utilisateur écrira l'URL pour laquelle il veut déterminer le caractère (pornographique ou sain)
- un bouton (Déterminer) qui lance l'analyse de l'URL
- des boutons (OK) et (ANNULER) qui permettent de quitter le programme
- une case à cocher (Mode Verbose) permettant d'afficher ou non les caractéristiques détaillées du site
- trois zones de texte dans lesquelles s'affiche l'état de l'analyse
- un menu qui sera détaillé plus loin

La figure 5.20 présente cette boite de dialogue.

5.7.2.2 *Menu de la boîte de dialogue principale*

Le menu se compose de trois éléments :

- Fichier : Quitter : permette de quitter le programme.
- Configuration :
  - Configuration : ouvre la boîte de dialogue de configuration, qui permet de configurer le filtrage (texte et/ou images, niveau de filtrage, algorithmes).
  - Mot de passe : permet de modifier le mot de passe protégeant l'accès à la boîte de dialogue de configuration
  - Dictionnaire : permet d'accéder à la boîte de dialogue dictionnaire pour ajouter d'autres mots au dictionnaire.

Figure 5. 20. *Interface de WebGuard*

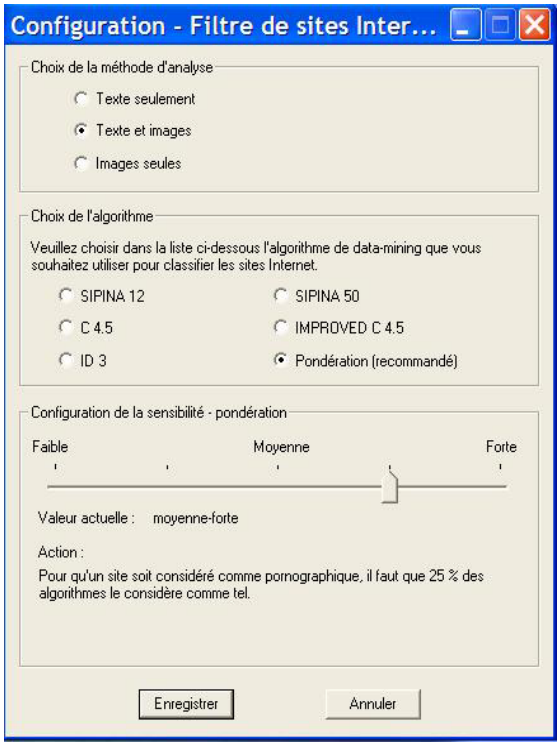


Figure 5. 21. *Boîte de dialogue de configuration*

La boîte de dialogue de configuration présenté par la figure 5.21 permet de paramétrer le filtrage. Elle est composée de trois parties :

1. Le choix de la méthode parmi trois :
   - Texte seulement pour n'utiliser que les critères textuels de la page ;
   - Image seulement pour n'utiliser que les images présentes sur la page pointée par l'URL ;
   - Texte et image pour utiliser la combinaison textuelle/images, ce qui constitue l'originalité de notre programme.
2. Le choix de l'algorithme parmi les 5 utilisés :
   Il est possible de choisir les algorithmes séparément grâce à cinq choix (boutons radio) mais ce qui est plus intéressant est le fait que l'on puisse les appliquer tous les 5 en même temps grâce au système de pondération qui donne à chacun d'entre eux un poids relatif à son efficacité.

3. La configuration de la sensibilité lorsque la pondération est utilisée. Nous avons prévu trois niveaux :
    - Filtrage faible : pour une utilisation adulte, il faudra que les cinq algorithmes soient d'accord pour que le filtrage soit activé.
    - Filtrage moyen pour une utilisation familiale, c'est le filtrage que nous jugeons le plus utile
    - Filtrage élevé, niveau paranoïaque. A ce niveau, pratiquement tous les sites pornographiques seront bloqués, un seul logiciel déclare le site comme pornographique, le site est bloqué.

La sécurité de l'application est assurée par l'affichage d'une fenêtre demandant le mot de passe lors de l'activation du module de configuration, qui bien entendu ferme celui-ci si le texte saisi n'est pas celui sauvegardé précédemment (cf. figure 5.22).

Figure 5. 22. *Fenêtre pour accéder à la boîte de Configuration*

Le mot de passe est enregistré dans un fichier annexe et il est cryptés selon une méthode d'addition de la valeur au format ASCII de la lettre à coder avec celle d'un mot de code de huit lettres avec une permutation circulaire des lettres dans des sous-tableaux du tableau de codage ASCII. Cela nous permet d'être certain que personne ne pourra connaître le mot de passe sans une bonne dose d'efforts et un bon matériel informatique (évidemment cette technique n'est pas inviolable, aucune ne l'est réellement). Dans le cas de la suppression du fichier de configuration, le mot de passe n'existe plus donc on ne peut plus lancer l'application de configuration, et dans l'absence de valeur, le logiciel d'autorisation renvoie la réponse la plus stricte possible. Le client devra alors contacter la société pour demander une nouvelle version du fichier manquant.

Un point sur lequel nous attirons l'attention est qu'il faut faire attention et prendre des précautions pour que l'on ne puisse pas faire sauter le verrou en réinstallant le logiciel. Pour cela, il faut que le module d'installation vérifie que le programme n'a pas déjà été installé sur l'ordinateur et qu'une personne tente de refaire une installation pirate pour configurer le logiciel à son propre goût. Cela peut se faire par une inscription d'une donnée dans la base des registres qui ne sera effacée que dans le cas d'un Uninstall lancé par le logiciel avec le mot de passe en vigueur.

Enfin, nous signalons que nous avons envisagé trois types d'architecture réseau pour notre solution WebGuard : une solution tout client, une autre tout serveur et la dernière une solution hybride (voir annexe).

5.8 Conclusion

Dans ce chapitre, nous avons présenté notre solution de classification et de filtrage de sites à caractère pornographique, WebGuard qui s'appuie sur une approche d'apprentissage par des techniques d'extraction de connaissance et une analyse conjointe du contenu textuel, structurel et visuel. Les résultats de cette étude montrent que WebGuard atteint des performances supérieures aux logiciels existants sur le marché. WebGuard affiche sur notre base de test MYL un taux de classification de 96,1% quand seule une analyse du contenu textuel et structurel est utilisée (WebGuard-TS) et un taux de classification de 97,4% lorsque l'analyse du contenu visuel (WebGuard-V) a été également utilisée en cascade avec WebGuard-TS.

Les résultats expérimentaux sur une liste noire significative de 12311 sites adultes fournis par le ministère de l'éducation Française ont montré l'efficacité et l'apport de l'analyse du contenu visuel pour la détection et le filtrage des sites adultes. En effet, en utilisant WebGuard-TSV nous avons obtenu un taux de classification de 95,62%, alors qu'une analyse basée uniquement sur le contenu textuel et structurel par WebGuard-TS n'affiche qu'un taux de classification de 87,82%.

Chapitre 6

Conclusion et perspectives

La définition d'un modèle de peau permettant d'identifier les pixels de peau dans les images a de nombreuses applications : détection et reconnaissance de visages, filtrage de sites pornographiques, etc. Dans cette thèse nous avons proposé une approche d'extraction de connaissance pour l'élaboration d'un modèle de peau avec application notamment sur le filtrage de sites web. Basé sur la couleur qui traduit la luminosité réfléchie par une surface de peau, notre modèle de peau se distingue de nombreux travaux de la littérature par sa *généricité* qui résulte d'une part d'une sélection de variables pertinentes des axes de couleur qui font appel notamment à la distribution spectrale de couleur, et d'autre part à des corpus significatifs de pixels de couleur traduisant la diversité de conditions de lumière et la richesse d'ethnies. L'originalité de nos travaux réside aussi dans l'utilisation conjointe d'une technique de segmentation basée sur la croissance de région pour le filtrage du bruit qui peut subsister après l'application de notre modèle de peau sur une image.

Afin d'illustrer la robustesse et la généricité de notre approche, le modèle de peau issu de nos travaux a été utilisé avec succès dans trois applications. La première application concerne la détection de visages dans les images vidéo. L'utilisation de notre modèle de peau a permis un gain en taux de détection de visages par rapport au modèle de peau issu d'une approche bayesienne. La deuxième application s'inscrit dans le cadre du projet national RNTL Muse visant un moteur de recherche multimédia et elle concerne la classification d'un portrait en gros plan, plan américain et plan en pied.

La troisième application est la classification et le filtrage de sites adulte qui est également l'objet de notre chapitre 5. Nos travaux en la matière ont été motivés par la prolifération sur le Web de contenus préjudiciables comme par exemple la pornographie, contenus préjudiciables notamment pour les enfants dont l'accès à l'Internet devient quotidien. A la différence de la majorité de systèmes commerciaux basés essentiellement sur la détection de mots indicatifs ou l'utilisation d'une liste noire manuellement collectée, notre système, WebGuard, s'appuie sur un apprentissage automatique par des techniques de data mining et une analyse du contenu textuel et structurel combinée à une analyse du contenu visuel basé sur la couleur de peau. WebGuard atteint un taux de classification de 97,4% sur 400 sites composés de 200 sites adultes et 200 sites non adultes. Expérimenté sur une liste noire de 12 311 sites manuellement classifiés par le Ministère de l'Education Nationale, WebGuard atteint un taux de classification de 95,62%. La figure 6.1 illustre schématiquement les étapes suivies et les applications réalisées.

En résumé, les techniques d'extraction de connaissance ont été au centre de nos travaux, que ce soit pour la définition d'un modèle de peau ou ses applications. Les perspectives en sont multiples. Nous les développons dans la suite autour de deux axes.

D'abord, sur l'amélioration d'un modèle de peau. Dans nos travaux, la définition du modèle de peau est basée sur la construction d'un arbre de décision à partir d'un corpus significatif. L'apprentissage du modèle ne prend pas en compte l'information spatiale d'un pixel potentiellement de peau par rapport à son voisinage. Cette information du voisinage n'est prise en compte qu'en une deuxième étape lors de la segmentation de régions de peau par la croissance de région. Or, cette information peut être prise en compte dans le modèle de peau dès l'apprentissage, en comptant par exemple la distance d'un pixel à ses voisins comme une nouvelle caractéristique. Une telle approche risque d'alourdir la phase d'apprentissage

mais il serait intéressant de comparer les résultats de cette approche avec ceux obtenus par notre modèle de peau et la prise en compte de l'information spatiale par une technique de segmentation de régions.

Ensuite, sur la classification et le filtrage de sites Web. Dans nos travaux, nous avons proposé une première stratégie d'intégration de l'analyse du contenu visuel, basé sur la couleur de peau ici, pour la classification et le filtrage de sites Web qui deviennent de plus en plus visuels et multimédias. Les résultats ont été largement concluants. D'un taux de classification de 87,82% sur une liste noire de 12311 sites du Ministère de l'Education Nationale à partir d'une analyse du contenu textuel et structurel, l'intégration de l'analyse du contenu visuel dans WebGuard a permis d'atteindre un taux de classification de 95,62%. Ces résultats encourageants nous incitent à approfondir nos travaux de classification et filtrage de sites Web par une analyse conjointe de plusieurs modalités et à les appliquer à d'autres problèmes comme par exemple le filtrage de sites xénophobes, racistes, etc. Si une intégration fine de l'analyse du contenu visuel permet d'améliorer le taux de classification, on peut aussi penser que le contenu textuel et structurel d'une page Web permet aussi d'aider à la classification d'images ou titres musicaux sur le Web. D'autres pistes d'amélioration concernent l'élaboration du dictionnaire des mots clés qui a joué un rôle central dans les performances de WebGuard. Or, l'élaboration de ce dictionnaire, qui contient actuellement plus de 400 mots dans plusieurs langues, a été très laborieuse car manuellement améliorée étape après étape, et elle n'a vraisemblablement possible que grâce à la compréhensibilité des modèles obtenus par les techniques d'extraction de connaissance. Il serait donc intéressant d'automatiser par l'apprentissage à partir d'un corpus la construction d'un tel dictionnaire pour qu'il ne contienne des mots ou des phrases indicatives permettant de discriminer les classes. La définition de mots ou phrases utilisés comme descripteur de contenu textuel peut elle aussi être améliorée pour prendre en compte par exemple les variations morphologiques d'un mot, les combinaisons de ceux-ci, par des n-grammes par exemple, ou encore à un niveau ultime le sens même d'une phrase, péjorative ou non, etc.

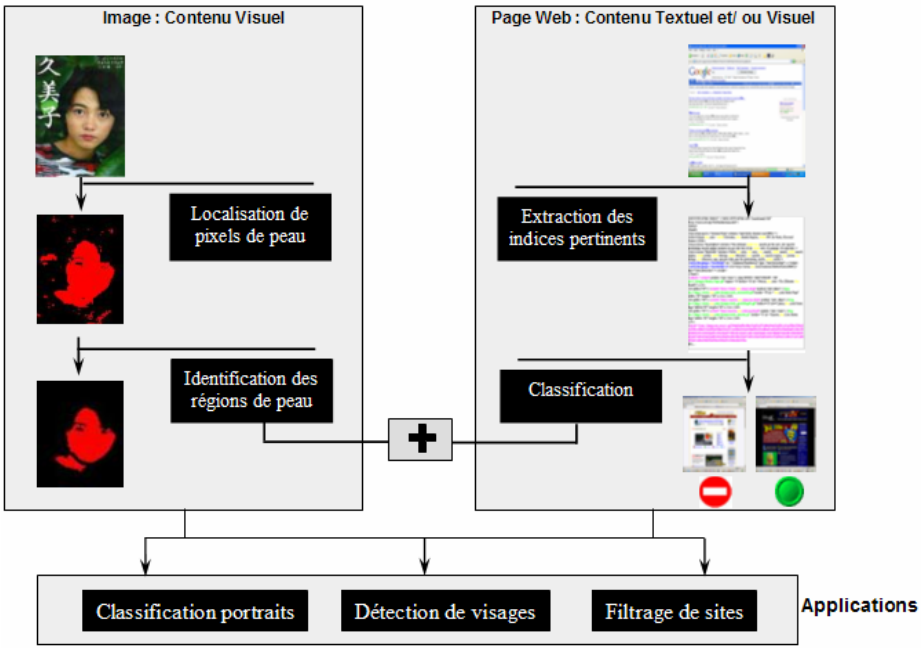


Figure 6.1. *Etapes suivies et applications réalisées*

Bibliographie

[1]. M. Hammami, L. Chen, "Web adult content detection and filtering system", International Journal of Business Data Communications and Networking. ISSN: 1548-0631.

[2]. M. Hammami, Y. Chahir, L. Chen, D. Zighed, " Détection des régions de couleur de peau dans l'image", revue RIA-ECA, vol 17, Ed.Hermès, ISBN 2-7462-0631-5, pp. 219-231, Janvier 2003.

[3]. M. Hammami, L. Chen, D. Zighed, Q. SONG, « Définition d'un modèle de peau et son utilisation pour la classification des images, Ed. Hermès, ISBN 2-7462-0500-9, pp. 186-197, Juin 2002.

[4]. D. Tsishkou, M. Hammami, L. Chen, "Face Detection in Video Using Combined Data-mining and Histogram based Skin-color Model", IEEE Third International Symposium on Image and Signal Processing and Analysis, IEEE Catalog Number 03EX651, ISBN 953-184-062-8, Rome, Italy, pp. 500-503, September 18-20, 2003.

[5]. M. Hammami, D. Tsishkou, L. Chen, "Data-mining based Skin-color Modeling and Applications", Third International Workshop on Content-Based Multimedia Indexing, Ed. SuviSoft Oy Ltd, ISBN 2-7261-1254-4, Rennes, France, pp. 157-162, Septembre 22-24, 2003.

[6]. M. Hammami, Y. Chahir, L. Chen, "WebGuard : Web Based Adult Content Detection and Filtering System", The 2003 IEEE/WIC International Conference on Web Inteligence, IEEE Computer Society Order Number PR01932, ISBN 0-7695-1932-6, Halifax, Canada, Octobre 13-17, pp. 574-578, 2003.

[7]. M. Hammami, Y. Chahir, L. Chen, "Combining Text and Image Analysis in The Web Filtering System: WebGuard", IADIS International Conference: WWW/Internet 2003, ISBN 972-98947-1-X, Algarve, Portugal, November 5-8, pp. 611-618, 2003.

[8]. M. Hammami, D. Tsishkou, L. Chen,"Data-mining based Skin-color Modeling using the ECL Skin-color Images Database", International Conference on Computational Science ICCS'2004, Springer Verlag Editor, ISBN 3-540-22114-X, Krakow, Poland, Proceedings, Part I, pp. 310-317, June 6-9, 2004.

[9]. B. Ben Amor, M. Hammami, C. Vial, L. Chen, "The ECL image indexing tool and his integration in RNTL MUSE Project", Second International Conference on Intelligent Access to the Multimedia Documents on the Internet (MediaNet'2004), Tozeur, Tunisia, pp. 67-78, November 25-28, 2004.

[10]. M. Hammami, D. Tsishkou, L. Chen, "Adult Content Web Filtering and Face Detection using Data-mining based Skin-color Model", IEEE International Conference on Multimedia and Expo 2004 (ICME), IEEE Catalog Number: 04TH8763C, ISBN: 0-7803-8604-3, Taipei, Taiwan, June 27-30, 2004.

[11]. M. Hammami, L. Chen, B. Ben AMOR, C. Vial, "Classification d'images par concept ", Ed. Hermès, ISBN 2-7462-0500-9, pp. 379-385, Juin 2002.

[12]. M. Hammami, B. Ben AMOR, L. Chen " Classification automatique d'images", Revue des Nouvelles Technologies de l'Information (RNTI), Vol 2, Ed.Cépaduès, ISBN 2.85428.636.7, pp.278, Janvier 2004.

[13]. J. C. Terrillon, M. Shirazi, H. Fumakachi and S. Akamatsu, Comparative performance of different skin chrominance modelsand chrominances spaces for the automatic detection of human faces in color images, in Proc. IEEE International Conference on Face and Gesture Recognition, Grenoble, pp. 54-61, March 2000.

[14]. R. Brunelli and T. Poggio, Face recognition: features versus templates, IEEE-T-PAMI, Vol. 15, No. 10, pp.1042-1052, Octobre 1993.

[15]. A. Pentland, B. Moghaddam and T. Starner, View-based and modular eigenspaces for face recognition, IEEE-C-CVPR, Seattle, WA, USA, pp. 84-91, 1994.

[16]. N. Mottin, Localisation de visages dans des images acquises avec différents cadrages de caméras. Application à l'indexation, DEA ARAVIS (image et vision) Université Sophia Antipolis, Juin 2000.

[17]. D. CHAI and King. N. Ngan, Face Segmentation Using Skin-Color Map in Videophone Applications, IEEE Transactions on Circuit and Systems For video Technology, Vol. 9, N° 4, Juin 1999.

[18]. S. Marcel, O. Bernier et D. Collobert, Approche EM pour la construction de régions de teinte homogènes : application au suivi du visage et des mains d'une personne, Université de Poitiers, CORESA 2000, Octobre 2000.

[19]. K. Schwerdt and J. L. Crowley, Robust Face Tracking using Color, in Proceeding of the 4th IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble, Mars 2000.

[20]. J. Ahlberg, Extraction and Coding of Face Model Paramaters, Licentiate Thesis No. 747, Departement of Electrical Engineering, Linlkoping University, Sweden, March 1999.

[21]. K. Sobottka and I. Pitas, Extraction of Facial Regions and Features using Color and Shape Information, 1997-2001 NEC Research Institue, 1996.

[22]. L. Gareth, H. Eunjung and O. Rolyn, A 3 Head Tracker For an Automatic Lipreading System, in Proc. of Australian Conference on Robotics and Automation, ACRA2000, Melbourne, Australian, August 2000.

[23]. J. Yang and A. Waibel, Tracking Human Faces in Real-Time, School of Computer Science, Technical report, Carnegie Mellon University, November, 1995.

[24]. V. Girondel, Détection de peau, suivi de tête et de mains pour des applications multimédia, DEA Signal Image Parole Télécom, Juillet 2002.

[25]. P. Wellner, «The DigitalDesk Calculator: Tactile Manipulation on a desktop». Dans

ACM Symposium on User Interface Software and Technology, Novembre 1991, pp. 27-33.

[26]. P. Wellner, « Interacting with Paper on the DigitalDesk». Rapport technique EPC-93-110, EuroPARC, Xerox Center, 1993.

[27]. O. Chomat. « Caractérisation d'éléments d'activités par la statistique conjointe de champs réceptifs » Thèse de doctorat, Institut National Polytechnique de Grenoble, Juin 2000.

[28]. F. Bérard, « Vision par ordinateur pour l'interaction homme-machine fortement couplée », thèse de doctorat, Université Joseph Fourier, Grenoble, France, Janvier 2000.

[29]. M. J. Swain and D.H. Ballard, « Color Indexing ». International Journal of Computer Vision, 1998.

[30]. A. Lemieux and M. Parizeau, Flexible multi-classifier architecture for face recognition systems. Vision Interface, 2003.

[31]. A. Elgammal, D. Harwood and L. Davis, Non-parametric model for background subtraction. European Conference on Computer Vision, pp. 751–767, 2000.

[32]. M. Alan, McIvor and Background subtraction techniques. IVCNZ, 2000.

[33]. M. Talibi Alaoui, R. Touahni et A. Sbihi, Classification des Images Couleurs par association des Transformations Morphologiques aux Cartes de Kohnen, CARI 2004, pp.83-90.

[34]. R. Cucchiara, C. Grana, M. Piccardi, A. Pratti and S. Sirotti: Improving shadow suppression in moving object detection with hsv color information. Intelligent Transportation Systems, pp. 334– 339. IEEE, 2001.

[35]. T. Horprasert, D. Harwood and L. Davis, A robust background subtraction and shadow detection. In Proceedings of the ACCV, 2000.

[36]. M. Ivana Mikiæ, Pamela C. Cosman, Greg T. Kogut et Mohan M. Trivedi : Moving shadow and object detection in scenes. In International Conference on Pattern Recognition (ICPR), pp. 321–324, 2000.

[37]. K. Toyama, J. Krumm, B. Brumitt and B. Meyers, Wallflower: Principles and practice of background maintenance. In Proceedings of the International Conference on Computer Vision (ICCV'99), pp. 255–261, 1999.

[38]. C. R. Wren, A. Azarbayejani, T. Darrell and A. Pentland, Pfinder : Real-time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7), pp. 780–785, 1997.

[39]. A. Wu, M. Shah, and N. da Vitoria Lobo. A virtual 3d blackboard: 3d finger tracking

using a single camera. In Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), March 2000.

[40]. A. Cavallaro and T. Ebrahimi. Accurate video object segmentation through change detection. In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'02), pp. 445–448, August 2002.

[41]. M. Harville, G. G. Gordon, and J. Woodfill. Foreground segmentation using adaptive mixture models in color and depth. In IEEE Workshop on Detection and Recognition of Events in Video, pp. 3–11, 2001.

[42]. Y. Ivanov, A. Bobick, and J. Liu. Fast lighting independent background subtraction. International Journal of Computer Vision, 37(2), pp. 199–207, 2000.

[43]. P. Wayne Power and Johann A. Schoonees. Understanding background mixture models for foreground segmentation. In Proceedings Image and Vision Computing New Zealand, November 2002.

[44]. P. L. Rosin and T. Ellis. Image difference threshold strategies and shadow detection. In 6th British Machine Vision Conference, pp. 347–356, 1995.

[45]. C. C. Chiang, W. N. Tai, M. T. Yang, Y. T. Huang and C. J. Huang. A novel method for detecting lips, eyes and faces in real time. Real-Time Imaging, 9, 2003.

[46]. R. Séguier, Détection de visage adaptative, 9èmes journées d'études et d'échanges "Compression et Représentation des Signaux Audiovisuels" CORESA 2004, Lile, Mai 2004.

[47]. V. Govindaraju, D. B. Sher, R.K. Srihari et S.N. Srihari, 'locating human faces in newspaper photographs', Proc. Conf. On Comp. Vision and Pattgern Recognition, pp. 549-554, 1989.

[48]. J. B. Waite and W.J. Welsh, "an application of active contour models to head boundary location", Proc. British Machine Vision Conf., pp. 407-412, Oxford, 1990.

[49]. I. Craw, D. Tock et A. Bennett," Finding face features", Proc. 2nd European Conference on computer Vision, pp. 92-96, 1992.

[50]. T. F. Cootes, C. J. Taylor, D.H. Cooper and J. Graham, "Active shape models-their training and application", comp. Vision and Image Understanding, Vol. 61, No. 1, pp. 38-59, 1995.

[51]. A. Jacquin and A. Eleftheriadis, "Automatic location tracking of faces and facial features in video signal", Int. Work. On Automatic Face and Gesture Recognition, pp. 142-147, Zurich, 1995.

[52]. T. Kanade, "Picture processing system by computer complex and recognition of humain faces", Phd thesis, Departement of Information Science, Kyoto University, Novembre 1973.

[53]. G. J. Klinker, S.A. Shafer and T. Kanade, “A physical approach to color image understanding“, International Journal of Computer Vision, Vol. 4, pp 7-38, 1990.

[54]. Y. Sumi et Y. Ohta, “Detection of face orientation and facial components using distributed appearance modelling”, Proc. Int. Work. on Automatic Face and Gesture Recognition, pp. 254-259, Zurich, 1995.

[55]. A. Zelinksy and J. Heinzmann, “Real time visual recognition of facial gestures for human computer interaction”, 2nd Int. Conf. On Automatic Face and Gesture Recognition, pp. 351-356, Vermont, 1996.

[56]. T. Leung, M. Burl et P. Perona, ‘Finding faces in cluttered scenes using labelled random graph matching’, Proc. 5th Int. Conf. on Comp. Vision, pp. 637-644, Cambridge, 1995.

[57]. H. P. Graf, E. Cosatto, D. Gibbon, M. Kocheisen et E. Patajan, “Multimodal system for locating heads and faces’, Proc.2nd Int. Conf. On Automatic Face and Gesture Recognition, pp. 88-93, Vermont, 1996.

[58]. M. C. Burl and P. Perona, ‘Recognition of planar object classes’, Comp. Vision and Pattern Recognition, San Fransisco, Juin 1996.

[59]. K. C Yow and R. Cipolla, “Finding initial estimates of humain face location”, Departement of Engineering, university of Cambridge, 1995.

[60]. B. Schiele and A. Waiebel, ‘Gaze tracking based on face color’, Proc. Int. Work. on automatic Face and Gesture Recognition, pp. 344-349, Zurich, 1995.

[61]. K. C. Yow et R. Cipolla, “Enhancing human face detection using motion and active contours”, Proc. 3rd Asian Conf. on Comp. Vision, Vol. 1, pp. 515-522, Hong Kong, 1998.

[62]. K. Aas, ”Detection and recognition faces in video sequences”, Norsk regnesentral, 1998.

[63]. R. Kjeldsen and J. kinder, “Finding skin in color images”, Proc. 2nd Int. Conf. on Automatic Face and Gesture Recognition, pp. 312-318, Vermont, 1996.

[64]. Q. Chen, H. Wu and M. Yachida, « face detection by fuzzy pattern matching », Proc. 5th Int. Conf. on Computer Vision, pp. 591-596, Cambridge, 1995.

[65]. Y. Dai and Y. Nakano, “Face texture model based on SGLD and its application in face detection in a color scene”, Pattern recognition, Vol. 29, N°.6, pp. 1007-1017, 1996.

[66]. M. Hunke and A. Waibel, “Face locating and tracking for humain computer interaction”, IEEE Computer, pp. 1277-1281, November 1994.

[67]. J. Yang, W. Lu, and A. Waibel, « Skin color modeling and adaptation’, in proceeding of ACCV’98, Vol.2, pp. 687-694, Hong Kong, 1998.

[68]. R-L. Hsu, M. Abdel-Mottaleb and A.K. Jain. Face detection in color images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5), pp. 696-706, May 2002.

[69]. E. Saber and A. Tekalp. Frontal-view face detection and facial feature extraction using color, shape and symmetry based cost functions. Pattern Recognition Letters, 19(8), pp. 669-680, 1998.

[70]. M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. Technical Report Cambridge Research Laboratory, CRL 98/11, Compaq, 1998.

[71]. J. C. Terrillon, M. N. Shirazi, H. Fukamachi, and S. Akamatsu. Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. In Fourth International Conference on Automatic Face and gesture Recognition, pp. 54-61, 2000.

[72]. M. Yang and N. Ahuja. Detecting human faces in color images. International Conference on Image Processing (ICIP), Chicago, pp.127– 130, 1998.

[73]. M. R. Teague, Image analysis via the general theory of moments. Journal of the Optical Society of America, 70(8), pp. 920–930, 1980.

[74]. E. Parzen, "On estimation of a probability density function and mode", Ann. Math. Stat., Vol.33, pp. 1065-1076, 1962.

[75]. R.O. Duda and P.E. Hart, "Pattern classification and scene analysis", John Wiley, 1973.

[76]. C.M. Bishop, "Neural Networks for Pattern Recognition", Clarendon Press, Oxford, 1995.

[77]. V. Vezhnevets, V. Sazonov, and A. Andreeva. A survey on pixel-based skin color detection techniques. In Proc. 13th International Conference on the Computer Graphics and Vision, Moscow, Russia, pp. 85-92, September 2003.

[78]. J. Brand and J.S. Mason. A comparative assessment of three approaches to pixel-level human skin-detection. In Proceedings. 15th International Conference on Pattern Recognition, vol. 1, Barcelona, Spain, pp. 1056-1059, September 2000.

[79]. G. Gomez. On selecting colour components for skin detection. In Proceedings. 16th International Conference on Pattern Recognition, vol. 2, pp. 961-964, 2002.

[80]. M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. International Journal of Computer Vision, 46(1), pp. 81-96, January 2002.

[81]. D. Chai and A. Bouzerdoum. A bayesian approach to skin color classification in YCbCr color space. In Proc. IEEE Region Ten Conference (TENCON'2000), Vol. 2, pp. 421-424, 2000.

[82]. B. D. Zarit, B.J. Super, and F. K. H. Quek. Comparison of five color models in skin pixel classification. In Proceedings. International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, Corfu, Greece, pp.58-63, September 1999.

[83]. D. Brown, I. Craw, and J. Lewthwaite. A som based approach to skin detection with application in real time systems. In Proc. of the British Machine Vision Conference, volume 2, pp. 491-500, 2001.

[84]. P. Peer, J. Kovac, and F. Solina. Human skin colour clustering for face detection. International Conference on Computer as a Tool, EUROCON 2003, Ljubljana, Slovenia, September 2003.

[85]. L. Jordao, M. Perrone, J.P. Costeira, and J. Santos-Victor. Active face and feature tracking. In Proceedings of International Conference on Image Analysis and Processing, pages 572-576, Venice, Italy, September 1999.

[86]. J. Z. Wang, J. Li, G. Wiederhold, and O. Firschein. Classifying objectionable websites based on image content. Notes in Computer Science, Special issue on iteractive distributed multimedia systems and telecommunication services, pp.113-124, 1998.

[87]. J. Z. Wang, J. Li, G. Wiederhold, and O. Firschein. System for screening objectionable images. Images, Computer Communications Journal, pp. 1355-1360, 1998.

[88]. L. Sigal, S. Sclaroff, and V. Athitsos. Skin color-based video segmentation under time-varying illumination. IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 862-877, July 2004.

[89]. S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Santa Barbara, CA, pp. 232-237, 1998.

[90]. L. Jordao, M. Perrone, J. P. Costeira, and J. Santos-Victor. Active face and feature tracking. In Proceedings of International Conference on Image Analysis and Processing, Venice, Italy, pp. 572-576, September 1999.

[91]. D. Saxe and R. Foulds. Toward robust skin identification in video images. 2nd International Face and Gesture Recognition Conference, Septembre 1996.

[92]. M. M. Fleck, D. A. Forsyth and C. Bregler. Finding naked people. European Conference on Computer Vision, Springer-Verlag, Berlin, Germany, pp. 592-602, 1996.

[93]. D. A. Forsyth and M. M. Fleck. Identifying nude pictures. IEEE Workshop on the Applications of Computer Vision, pp. 103-108, 1996.

[94]. H. Zheng, H. Liu and M. Daoudi. Blocking objectionable images: adult images and harmful sysbols. Proceedings of IEEE International Conference on Multimedia and

Expo, Taipei, Taiwan, June, 2004.

[95]. H. Wu, T. Yokoyama, D. Pramadihanto, and M. Yachida: Face and facial feature extraction from color images. International Conference on Automatic Face and Gesture Recognition, pp. 345-350, October 1996.

[96]. K. Sobottka, I. Pitas, Segmentation and tracking of faces in color images. Second International Conference on Automatic Face and Gesture Recognition, pp. 236-241, October 1996.

[97]. M. Collobert, R. Feraud, G. L. Tourneur, D. Bernier, J. E. Viallet, Y. Mahieux, and D. Collobert, Listen: A system for locating and tracking individual speakers. International Conference on Automatic Face and Gesture Recognition, pp. 283-288, October 1996.

[98]. J. Y. Lee and S. I. Yoo. An elliptical boundary model for skin color detection. In Proc. International Conference on Imaging Science, Systems and Technology, Las Vegas, USA, June 2002.

[99]. T. A. Mysliwiec, Fingermouse: A freehand computer pointing interface. Technical report, University of Illinois at Chicago, 1994.

[100]. F. K. H. Quek, T. Mysliwiec, and M. Zhao. Fingermouse: A freehand pointing interface. Proceedings of the International Workshop on Automatic Face- and Gesture-Recognition, Zurich, Switzerland, pp. 372-377, June 1995.

[101]. S. Ahmad: A usable real-time 3d hand tracker. Conference Record of the Asilomar Conference on Signals, Systems and Computers, pp. 1257-1261, 1994.

[102]. J. Cai, A. Goshtasby, and C. Yu. Detecting human faces in color images. Int'l Workshop on Multi-Media Database Management Systems, pp.124-131, 1998.

[103]. H. D. Cheng, X. H. Jiang, Y. Sun, and J. Wang. Color image segmentation: advances and prospects. Pattern Recognition, 34(12), pp. 2259-2281, December 2001.

[104]. M. Soriano, B. Martinkauppi, S. Huovinen, and M. Laaksonen. Skin detection in video under changing illumination conditions. In Proc. Computer Vision and Pattern Recognition, vol. 1, pp. 839-842, 2000.

[105]. A. Albiol, L. Torres, and E.J. Delp. Optimum color spaces for skin detection. In Proc. of the International Conference on Image Processing, vol. 1, Tessaloniki, Greece, pp. 122-124, 2001.

[106]. S. L. Phung, A. Bouzerdoum, D. Chai. A novel skin color model in ycbcr color space and its application to human face detection. In IEEE International Conference on Image Processing (ICIP'2002), vol. 1, pp. 289–292, 2002.

[107]. B. Menser, And M. Wien. Segmentation and tracking of facial regions in color image sequences. In Proc. SPIE Visual Communications and Image Processing 2000, pp. 731–740, 2000.

[108]. F. Marques, V. Vilaplana. A morphological approach for segmentation and tracking of human faces. In International Conference on Pattern Recognition (ICPR'00), vol. 1, pp. 5064–5068, 2000.

[109]. C. Wang, M. Brandstein. Multi-source face tracking with audio and visual data. In IEEE MMSP, pp.169–174, 1999.

[110]. R. Schumeyer, and K. Barner. A color-based classifier for region identification in video. In Visual Communications and Image Processing 1998, SPIE, vol. 3309, pp. 189–200, 1998.

[111]. L. Sigal, S. Sclaroff, and V. Athitsos. Estimation and prediction of evolving color distributions for skin segmentation under varying illumination. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 152–159, 2000.

[112]. M. J. Jones, and J. M Rehg. Statistical color models with application to skin detection. In Proc. of the CVPR '99, vol. 1, pp. 274–280, 1999.

[113]. S. Roux and E. Petit. 'Codeur H.263 amélioré par la visiophonie mobile, 7ème journée d'échange : Compression et représentation des signaux audiovisuels (CORESA), Dijon, Novembre 2001.

[114]. A. Trémeau et al. Image couleur : de l'acquisition au traitement. Collection Sciences Sup, 480 pages, dunod éditions, ISBN 2 10 006843 1, 2004.

[115]. R. O. Duda, P. E. Hart. « Pattern classification and scene analysis » John Wiley, 1973.

[116]. H. Ouhaddi, P. Horain, K. Milkolajczyk. Modélisation et suivi de la main. Actes 4èmes Journées d'études et d'échanges Compression et REprésentation des Signaux Audiovisuels (CORESA'98), Lannion, France, 9-10 June 1998, pp. 109-114, 1998.

[117]. D. G. Lowe. Robust Model-based Motion Tracking Through the Integration of Search and Estimation", International Journal of Computer Vision, 8:2, pp. 113-122, 1992.

[118]. F. Prêteux, P. Horain, H. Ouhaddi, and M. Preda. Report on Core Experiment 3 on Hand Baps interpretation. ISO/IEC JTC1/WG1 MPEG97/M3332, March 1998, http://www-sim.int-evry/Publications.

[119]. R. A. Fisher, The use of multiple measurements in taxonomic problems, Annals of Eugenics 7, pp. 179-188, reprinted in Contributions to Mathematical statistics, John Wiley, New-York, 1950.

[120]. T. Kohonen. Self-Organizing Maps. Springer Verlag, 1995.

[121]. T. Kohonen. Self-Organizing and Associative Memory. Springer Verlag, 2ème edition, New York, 1984.

[122]. R. Lefebure, G. Venturi, Data Mining. Paris, Eyrolles. 2001.

[123]. J. Clech, S. Hassas, Web Mining et système Multi-Agents. Tutoriel EGC 2003, Lyon, France

[124]. J. Clech, D. Zighed, Contribution Méthodologique à la Fouille de Données Complexes, thèse de doctorat, ERIC, 2004.

[125]. U. Fayyad, G. Piatetsky-Shapiro and P. Smyth, The KDD process for extracting useful knowledge from volumes data. Communication of the ACM 39 (11), pp. 27-34. 1996.

[126]. D. A. Zighed et R. Rakotomala, Extraction de connaissances à partir de données (ECD). Techniques de l'Ingénieur. HA. 2002.

[127]. D. A. Zighed et R. Rakotomala. A methode for non arborescent induction graphs. Technical report, Laboratory ERIC, University of Lyon2, 1996.

[128]. H. Simon, Why should machines learn? In R.S. Michalski, J.G. Carbonnel, and T.M. Mitchell, editors, Machine Learning: An Artificial Intelligence Approach. Morgan Kaufmann, 1983.

[129]. T. G. Dietterich, Learning at the knowledge level. Machine Learning, 1(3), pp. 287-316, 1986.

[130]. T. G. Dietterich and J. W. Shavlik, editors. Readings in Machine Learning. Morgan Kaufmann Publishers, Inc, 1990.

[131]. D.A.Zighed et R.Rakotomala, Graphes d'induction, Ed. Hermes, ISBN 2-7462-0072-4, 2000.

[132]. Y. Kodratoff, l'Extraction de connaissance à partir des données : un nouveau sujet pour la recherche scientifique, Apprentissage automatique, Ed. Hermes, ISBN 2-7462-0066-X, 1999.

[133]. Trun, The Monk's Problem: a performance comparaison of different learning algorithms. Technical Report CMU-CS-91-197, Carnegie Mellon University, 1991.

[134]. D. W. Aha, D. Kibler et M. K. Albert, Instance-based learning algorithms. Machine Learning 6(1). pp. 37-66.

[135]. W. Erray, G. Legrand, N. Nicoloyannis et D. Zighed, Sélection et Construction de variables, Rapport Interne, ERIC, janvier 2004.

[136]. L. Jourdan, G. Talbi et C. Dhaenens, Métaheuristiques pour l'extraction de connaissances: Application à la génomique, USTL, 2003.

[137]. C. Schaffer. Selecting a classification method by cross-validation. Machine learning, 13(1), pp. 135-143, 1993.

[138]. L. Prechelt. A quantitative study of experimental evaluation of neural network learning algorithm. Neural Networks, 9, 1996.

[139]. M. Stone. Cross-validatory choice and assement of statistical predictions. Journal of the Royal Statistical Society, pp. 111-147, 1974.

[140]. R. Kohavi and G. John. Wrappers for feature subset selection. AIJ Special Issue on relevance, pp. 273-324, 1997.

[141]. R. J. Brachman and T. Anand, The Process of Knowledge Discovery in Databases: A first Sketch, KDD 94, pp. 1-11, 1994.

[142]. M. Craven and J. W. Shavlik, Extracting tree-structured representations of trained networks. Advances in Neural Information Processing Systems, pp. 24-30, 1996.

[143]. R. S. Michalski. Theory and methodology of inductive learning. Machine learning: An Artificial Intelligence Approach, volume 1, pp. 83-134. Morgan Kaufmann, Los Atlos, 1983.

[144]. R. Andrews, J. Dietterich and A. B. Tickle. A survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-Based Systems, vol. 8, pp. 373-389, 1995.

[145]. M. Craven and J. W. Shavlik, Using sample and queries to extract rules from trained networks. In W.W Cohen and H. Hirsh, editors, Proceeding of the 11th International Conference on Machine Learning. Morgan Kaufmann, 1994.

[146]. D. H. Wolpert. Stacked generalization. Neural Networks, vol.5, pp. 241-259, 1992.

[147]. P. Clark. Machine learning: Techniques and recent developments. In A. R. Mirzai, Artificial Intelligence: Concepts and Applications in Engineering, Chapman and Hall, pp. 65-93, 1990.

[148]. J. R. Quinlan. Induction of decision trees. Machine Learning, 1:81-106, 1986.

[149]. J. R. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.

[150]. L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification of Regression Trees. Wadsworth, 1984.

[151]. I. C. Lerman, R. Gras et H. Rostam. Elaboration et évaluation d'un indice d'implication pour données binaires. Mathématiques et Sciences Humaines, pp. 5-35, 1981.

[152]. C. E. Shannon, ``A mathematical theory of communication," Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October, 1948.

[153]. Alain Chrisment, Couleur et Colorimétrie, Editions. 3C Conseil, ISBN 2-9508797-9-9, Paris 1997.

[154]. P. Callet, Couleur-lumière couleur-matière, Arts et Sciences. Ed. Diderot, 1998.

[155]. G. McGunnigle, The classification of textured surfaces under various illuminant direction, Ph.D. thesis, Heriot-Watt University, Departement of Computing and Electrical Engineering, June 1998.

[156]. L.T. Maloney, Color Vision, From Genes to Perception, chapter Physics-based approaches to modelling surface color perception, Cambridge University Press, 2000.

[157]. R. Hall, Illumination and Color in Computer Generated Imagery, Springer-Verlag, 1998.

[158]. G. J. Klinker, S. A. Shafer, and T. Kanade, A physical approach to color image understanding, International Journal of Computer Vision, vol 4, n°1, pp. 7-30, 1990.

[159]. S. Tominaga, Dichromatic reflection models for rendering object surfaces, Journal of Imaging Science and technology, vol. 40, n°6, pp. 549-555, December 1996.

[160]. G. Rougeron, Problèmes lies à la couleur en synthèse d'images, Ph.D. thesis, Ecole des mines de Saint-Etienne, Université Jean Monnet, 27 janvier 1998.

[161]. G.Wyszecki and W.S. Stiles, Color Sience : Concepts and Methods, Quantitative Data and Formulae, John Wiley & sons, 2nd edition, 1982.

[162]. D. L. MacAdam, Color Measurement, theme and variation, Optical Sciences. Springer-Verlag, second revised edition, 1985.

[163]. G. A. Agoston, Color Theory and its Application in art and design, Optical Science. Springer-Verlag, 1987.

[164]. P. Kowaliski, Vision et mesure de la couleur, Physique fondamentale et appliqué. Masson 2ème édition, 1990.

[165]. R. W. G. Hunt. Measuring Color, Applied science and industrial technology. Ellis Horwood, 2nd edition, 1991.

[166]. Robert sève, Physique de la couleur, Physique fondamentale et appliquée. Masson, 1ère édition, 1996.

[167]. R. Gershon, « Aspects of perception and computation in color vision » Computer Vision, Graphics, and Image Processing, vol.32, n°2, CVGIP, pp. 244-277, November 1985.

[168]. G. Sharma and H. J. Trussell, «Digital color imaging» IEEE Transaction on Image Processing, vol. 6, n° 7, pp. 901-932, 1997.

[169]. I. Kononenko, Estimating attributes: analysis and extensions of Relief. In L. De Raedt and F. Bergadano, editors, Machine Learning: ECML94, pp. 171-182. Springer Verlag, 1994.

[170]. F. Meyer, Skeletons in digital spaces. Image analysis and mathematical morphology,

theoretical advances. Serra. Academic press, 1988.

[171]. L. Vincent, P. Soille, Watershed in digital spaces, an efficient algorithm based on immersion simulation. Trans. PAMI vol 13, n° 6, 1991.

[172]. E. Karpova, D. Tsishkou and L. Chen "The ECL Skin-color Images from Video (SCIV) Database", in Proceeding of IAPR International Conference on Image and Signal Processing (ICISP'2003), Agadir, Maroc, pp.47-52, June 2003.

[173]. N. Vandenbroucke. Segmentation d'images couleur par classification de pixels dans des espaces d'attributs colorimètriques adaptés, Application à l'analyse d'images de football. Thèse de doctorat, Université Lille 1, Decembre 2000.

[174]. J. C. Russ. The Image Processing Handbook. CRC Press, Boca Raton, 1995.

[175]. Preston Gralla, Sherry Kinkoph « Internet et les enfants »Ed. CampusPress ISBN 2-7440-0979-2 pp. 74.

[176]. S. Chakrabarti, B. Dom, P. Indyk: Enhanced hypertext categorization using hyperlinks, Proceedings of the 1998 ACM SIGMOD international conference on Management of data.

[177]. P. Y. Lee, S. C. Hui, A. C. M. Fong, Neural Networks for Web Content Filtering, IEEE Intelligent Systems, pp. 48-57, Sept/oct, 2002.

[178]. Ann Beeson, Chris Hansen, «Farenheit 451.2: Is Cyberspace Buming ?».

[179]. P. Aftab, A Parent's Guide to the Internet, chap 16.

[180]. L. Breckelmans, With our Backs Against the Walls: Managing Acces to Internet Pornography in Libraries, chap. Proprietary Databases.

[181]. Hochheiser H., Filtering FAQ version 1.1, Computer Professionals for Social Responsibility, pt 2.1.

[182]. www.peacefire.org. Peacefire.org was created in August 1996 to represent the interests of people under 18 in the debate over freedom of speech on the Internet

[183]. Etude de NCAC (National Coalition against Censorship) sur le test d'une vingtaine d'outils de filtrage, Février 2003.

[184]. G. W. Flake, K. Tsioutsiouliklis, L. Zhukov, Methods for mining Web communities : bibliometric, spectral, and flow, Web Dynamics, editors A.Poulovassilis and M.Levene, Springer Verlag, 2003.

[185]. S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, in WWW7, Brisbane, Australia, 1998.

[186]. J. Cho, H. Garcia-Molina, L. Page, Efficient crawling through URL ordering,

Computer Networks and ISDN Systems, 30(1-7), pp. 161-172, 1998.

[187]. G. W. Flake, S. Lawrence, C. L. Giles, Efficient identification of web communities, in Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD-2000), Boston, MA, 2000, ACM Press.

[188]. Y. Yang, S. Slattery, R. Ghani, A study of approaches to hypertext categorization, Journal of Intelligent Information Systems, Kluwer Academic Press, 2001.

[189]. J. Fürnkranz, Exploiting structural information for text classification on the WWW, in Intelligent Data Analysis, pp. 487-498, 1999.

[190]. G. Attardi, A. Gulli, and F. Sebastiani, Automatic Web page categorization by link and context analysis, Proceedings of THAI-99, 1st European Symposium on Telematics, Hypermedia and Artificial Intelligence, pp. 105-119, Varese, IT, 1999.

[191]. E. J. Glover, K. Tsioutsiouliklis, S. Lawrence, D. M. Pennock, G. W. Flake, Using Web structure for classifying and describing Web pages, WWW2002, May 7-11, 2002, Honolulu, Hawaii, USA, 2002.

[192]. Recreational Software Advisory Council on the internet, association that became the Internet Content Rating Association (ICRA) in 1999, www.icra.org.

[193]. Cybersitter 2002 Copyright © 1995-2003, Solid Oak Software, Inc. All Rights Reserved. www.cybersitter.com.

[194]. Net Nanny 4.04 Copyright © 2002-2003 BioNet Systems, LLC. All Rights Reserved. www.netnanny.c om.

[195]. Norton Internet Security 2003 © 1995-2003 Symantec Corporation. All rights reserved. www.syman tec.com.

[196]. Puresight Home 1.6 iCognito Technologies Ltd. www.icognito.com.

[197]. Cyber Patrol 5.0 © 2003 SurfControl plc. All rights reserved. www.cyberpatrol

[198]. B.Stayrynkevitch, M.Daoudi, C.Tombelle, H.Zheng, et al., "Poesia Software architecture definition document", Technical report, Poesia consortium, December 2002.

Annexe

Architectures réseau envisageables Pour « WebGuard »

Nous avons envisagé trois types d'architecture réseau possibles pour notre technique:

- une solution tout client
- une solution tout serveur
- une solution hybride

A. La solution tout client

Cette solution consiste en l'exécution du programme filtrant sur l'ordinateur client, c'est-à-dire sur celui qui est utilisé pour accéder à Internet. Dans ce cas-là, lorsqu'une requête est faite par le navigateur, le programme filtrant détermine si l'URL demandée existe dans la liste noire (liste des sites pornographiques connus) ou dans la liste blanche (liste des sites reconnus comme sains). Si cette URL est présente dans une des deux listes, alors le programme laisse la requête se poursuivre ou non selon la couleur de la liste. Dans le cas contraire, l'analyse de la page sera faite. La figure A.1 montre qu'il y a trois chemins possibles: en vert, le site a déjà été visité et l'URL fait partie de la liste blanche; en rouge, le site a déjà été visité et l'URL fait partie de la liste noire; en jaune, le site n'a jamais été visité.

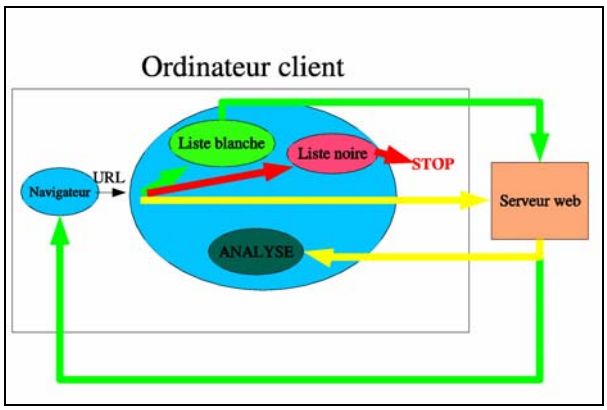


Figure A. 1. *Solution tout client*

Si l'Url n'appartient pas ni à la liste blanche ni à la liste noire WebGuard récupère et analyse le code source de la page, déclare si cette page autorisée ou non et enfin met à jour la liste noire et la liste blanche. La figure A.2 illustre cette démarche.

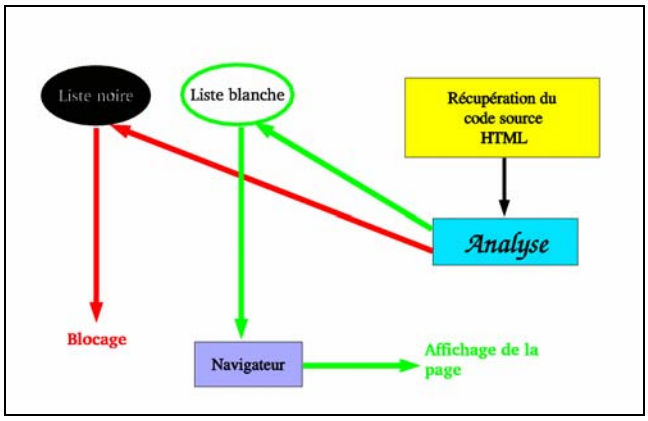


Figure A. 2. *Poursuite de la requête dans le cas d'analyse*

L'avantage de cette méthode est que chaque page est filtrée individuellement, le filtrage est par conséquent optimal. En revanche, cette méthode comporte un inconvénient majeur, provenant du fait que l'analyse s'effectue entièrement sur l'ordinateur client ce qui peut occasionner une gêne pour l'utilisateur. L'analyse n'est par instantanée, et même si l'analyse du code source est relativement rapide, l'analyse d'images l'est, quant à elle, beaucoup moins et très gourmande en mémoire et en pourcentage du CPU.

B. La solution tout serveur

Cette solution consiste à placer le logiciel filtrant sur un serveur, l'exécution se fera entièrement sur ce dernier. Il existe alors deux possibilités :

- L'exécution sur un serveur du fournisseur d'accès Internet (FAI)
- L'exécution sur un serveur dédié à cette tâche

Dans les deux cas, la recherche du caractère pornographique ou non du site demandé est faite par un ordinateur indépendant de l'ordinateur client. Dans cette solution, comme dans la solution précédente, le navigateur fait une requête et le programme filtrant détermine le type de site. La différence repose sur le fait que l'analyse s'effectue sur un serveur plus puissant et qui comporte plus d'espace de stockage et sur lequel donc, on peut avoir des listes blanches ou noires beaucoup plus longues. La mise en commun du serveur avec un grand nombre d'internautes permet un établissement de listes très fiables et très riches.

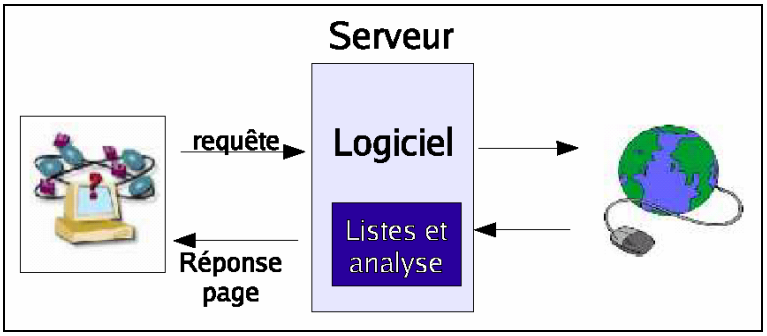


Figure A. 3. *Solution tout serveur*

Les avantages de cette solution sont principalement la longueur des listes et la durée d'analyse. En effet, le grand nombre d'internautes permet d'obtenir des listes aussi exhaustives que possible avec les sites les plus visités et l'analyse étant effectuée sur un serveur qui sera non seulement plus puissant que l'ordinateur client mais qui plus est exclusivement dédié à cette tâche, permettra d'obtenir une analyse très rapide.

Le principal inconvénient réside dans la mise en place de cette solution, plus difficilement réalisable de par, le coût d'un serveur et le peu d'intérêt qu'un FAI aurait à installer un tel logiciel.

C. La solution hybride

Cette solution combine les deux solutions précédentes avec la présence de deux modules : un module client et un module serveur.

- *Module client*

D'un côté, sur l'ordinateur client, se trouvent les listes des URLs noires ou blanches. Lors d'une requête, le programme compare l'adresse du site demandé avec celles présentes dans les listes blanches ou noires. Si l'adresse existe, le filtrage adéquat s'en suit. Le module client ne fait pas l'analyse des pages web, il ne fait qu'une comparaison et une recherche dans les listes. Ce module doit être configurable pour choisir le niveau de sécurité désiré, et proposer plusieurs profils d'utilisateurs, chacun ayant un niveau de sécurité propre. Il doit pouvoir aussi télécharger les mises à jour des listes sur le serveur.

- *Module serveur*

Le serveur scanne le web et procède aux analyses de chaque page téléchargée afin de grossir ses listes et de les mettre éventuellement à jour. L'utilisateur achète alors les mises à jour des listes d'URLs comme il achèterait des mises à jour de définitions de virus. Parmi les avantages de cette solution, on trouve la rapidité et la simplicité d'utilisation par le client du programme. Aucun ralentissement notable du à l'analyse, aucune manipulation compliquée, pour le client un simple téléchargement de mises à jour. De plus, le client peut choisir de ne conserver sur son disque dur que la liste noire, ce qui permette de minimiser la taille de l'espace disque utilisé. Le principal inconvénient réside dans le fait que certains sites récemment crées peuvent ne pas être filtrés si la mise à jour n'a pas été faite ou si le module serveur n'a pas eu le temps de visiter ce site.

Le tableau A.1 présente une synthèse des avantages et des inconvénients pour chaque une des trois solutions.

Tableau A. 1. *Avantages et Inconvénients de chaque solution*

Solution	Avantages	Inconvénients
Tout client	- Filtrage optimal	- Durées d'analyse longues - Ralentissement dans le processus d'affichage
Tout serveur	- Filtrage optimal - Listes beaucoup plus longues - Durées d'analyse très courtes	- Difficulté de mise en place - Manque d'intérêt d'un FAI dans la mise en place d'une telle solution
Solution hybride	- Simplicité et rapidité d'utilisation - Minimisation de l'espace disque utilisé - Listes aussi exhaustives que possible	- Difficulté pour gérer les sites « champignons »