Université de Nantes

Laboratoire d'Informatique de Nantes Atlantique

Structuration statistique de données multimédia pour la recherche d'information

Synthèse et travaux

présentés et soutenus publiquement le 30 novembre 2007

pour l'obtention de l'

Habilitation à Diriger des Recherches de l'Université de Nantes

Spécialité Informatique

par

Marc Gelgon

Composition du jury

Président :	José Martinez, Professeur à l'université de Nantes
Rapporteurs :	Michel Crucianu, Professeur au CNAM Mohand-Saïd Hacid, Professeur à l'université Claude Bernard, Lyon Claude Labit, Directeur de recherches INRIA
Examinateur :	Noureddine Mouaddib, Professeur à l'université de Nantes (direct. des travaux)

École Doctorale Sciences et Technologies de l'Information et des Matériaux

Mis en page avec la classe thloria.

Table des matières

1

Introduction

Ce document présente une synthèse de mes activités de recherche depuis la thèse.

L'unité du travail réside en ce qu'on s'intéresse à la recherche de structure dans les données numériques (issues de données multimédia), en vue d'y faciliter la recherche d'information. Le cadre méthodologique de la résolution est que nous privilégions ici celui des modèles probabilistes, en particulier les mélanges de lois, et de l'estimation statistique associée. La recherche de structure implique que le jeu de données étudié est composé de sous-populations de caractéristiques distinctes : il s'agit de séparer et de caractériser ces sous-populations, deux problèmes fortement imbriqués. Les entités extraites et les attributs qu'on en leur associe seront alors directement utiles pour la recherche d'information.

Dans cette introduction, je donne d'abord un bref point de vue personnel sur la manière dont la recherche par le contenu dans les données multimédia a évolué depuis ma thèse, puis l'organisation du document est décrite.

Il y a dix ans, la question de la recherche d'information dans les documents audiovisuels commençait son essor dans la communauté de recherche traitant de l'analyse automatique des contenus multimédia. Dans la suite de ce document, le mot "multimédia" ne fera pas référence à un document comportant nécessairement plusieurs média, mais est utilisé quand le propos s'applique, de façon générale, pour les média de type image, vidéo ou audio, dans la mesure où nombre de problèmes et de techniques "reconnaissance de formes" leur sont transversaux. J'ai commencé à m'intéresser à ce champ applicatif dans le début de la vague, grâce à mon directeur de thèse, Patrick Bouthemy. Ma thèse avait débuté à l'intention d'un partenariat avec un organisme de la Défense, concernant la segmentation spatio-temporelle de video au sens de la texture, puis s'est reconvertie vers l'application à l'indexation de video par le contenu. On trouvait là une perspective nouvelle et enthousiasmante, car probablement de grande portée, à terme, pour la manière dont toute la société accède à l'information. L'inflexion, à l'image que ce qui s'est produit dans de nombreux laboratoires de par le monde, a été facilitée par ce que nombre de tâches à réaliser et des manières de les accomplir (critères, modèles, algorithmes,...) s'appuieraient largement des techniques récemment élaborées pour

la vision par ordinateur (vision industrielle, bonne maîtrise de l'appariement et du lien 2D-3D acquis dans les années 90), la compression audiovisuelle ou l'authentification par la parole. Cette réutilisation a permis d'obtenir rapidement des résultats assez probants.

Après cette première phase, le domaine a parfois été caractérisé par des tâtonnements quant aux directions à prendre, en termes de fourniture de technologies d'analyse de contenu, largement parce que car le paysage applicatif, consommateur de ces technologies, était en principe prometteur mais, en pratique, plutôt brumeux.

Evoquons rapidement pourquoi la route n'était pas tracée d'avance : quels contenus - produits par qui ? Quels utilisateurs, sur quel terminal ? Quels acteurs industriels ? Quel modèles économiques et quelle utilité sociale ?

Un leitmotiv pour introduire un article dans le domaine était d'argumenter sur l'impossibilité d'étiqueter manuellement les données, pour justifier le besoin d'analyser automatiquement les données audiovisuelles. Malgré tout, dans de nombreux cas, un flou demeure concernant cette indisponibilité de méta-données, et le besoin d'analyser le contenu : on peut peut être se satisfaire des moteurs de recherche d'image exploitant le texte environnant, ou des sites grand public de mise en ligne de video reposant sur des méta-données introduites manuellement. L'étiquetage collaboratif entre utilisateurs est, dans certains cas, une nouvelle force ; de nouveaux capteurs (par ex. localisation géographique) viennent compléter les descriptions. La détection de copie par analyse du contenu est concurrencée par le tatouage, même s'ils peuvent être complémentaires.

Par ailleurs, étant donnée la "distance" entre les méta-données souhaitées par les applications et ce qu'on peut raisonnablement espérer extraire fiablement et automatiquement de signaux, le domaine a dû itérer entre ce qui est utile et ce qui est techniquement faisable, les deux étant dépendants et seulement partiellement observables. Enfin, dans la décennie passée, l'ensemble des utilisateurs de ces systèmes s'est élargi. Les partenaires industriels des contrats de recherche où j'étais impliqué en tant que doctorant (Institut National de l'Audiovisuel, chaînes de télévision) visaient plutôt les utilisateurs professionnels, et internet n'était pas au centre de ces questions. En dix ans, internet est entré chez les particuliers, devenus producteurs et chercheurs de contenus multimédia. Simultanément, le volume et la diversité des contenus mis en ligne connaissaient une croissance considérable ; il devenait possible de se libérer d'horaires de diffusion, en accédant de manière asynchrone aux contenus - produisant au passage une passionnante révolution de la manière dont la société peut s'informer. En résumé, le "paysage des contenus", qui conditionne pour partie les orientations de recherche appliquée dans le domaine, est complexe et changeant.

Une dernière difficulté est que des verrous en cours ont requis le démarrage de travaux réellement multi-disciplinaires : analyse des contenus et interaction homme-machine, analyse de contenus (éventuellement sous une forme encodée) et bases de données, analyse image et analyse audio.

Un (fort) second souffle a été trouvé dans ce domaine. Il a consisté en des objectifs applicatifs plus précis - recherche de vues issues de la même scène réelle, plutôt que d'images "similaires" dans un sens parfois peu défini - et économiquement motivés (détection de copie d'image ou de vidéo (Berrani, Amsaleg & Gros 2003, Poullot, Buisson & Crucianu 2007),

spécialisation dans certains contenus à la fois "rentables" et permettant de forts a priori dans l'analyse : sports (Kokaram, Rea, Dahyot, Tekalp, Bouthemy, Gros & Sezan 2006) ou journaux télévisés (Wactlar, Kanade, Smith, Stevens & S.Smith 1996).

Le travail présenté dans ce document a tenté de s'inscrire dans cette "seconde vague" des travaux en indexation multimédia : d'une part, en s'intéressant au cas des données multimédia personnelles, ensuite au lien entre structuration de données multimédia et bases de données.

Dans le **chapitre 2, mélanges de lois**, nous présentons les modèles probabilistes de type *mélange de lois* et algorithmes d'estimation associés, parce qu'ils sont la colonne vertebrale de la plupart des travaux que nous présentons dans ce document. Quantité de variantes découlent de ces modèles, tant en termes de structure de modèle graphique que d'algorithmes d'estimation ; la diversité des applications auxquelles ces variantes peuvent répondre est remarquable, dans tous les domaines du multimédia. Enfin, au nombre de leurs qualités, j'ai plusieurs fois observé auprès d'étudiants que les mélanges de lois ouvrent nombre de verrous pédagogiques.

La fig. 1 illustre ma trajectoire depuis la thèse :

- mon travail de doctorat (1995-1998) a concerné l'analyse d'images numériques, je l'ai réalisé dans les projets INRIA TEMIS puis VISTA, à l'IRISA, Rennes. Cette recherche a concerné l'estimation/segmentation de mouvement 2D dans des séquences d'image et le suivi temporel d'objet. Les outils employés étaient les modèles markoviens, l'estimation et les tests statistiques, et le domaine applicatif, l'indexation de video.
- de fin 1998 à 2000, j'ai travaillé au centre de recherche de Nokia, à Tampere, Finlande, alors qu'on commençait à embarquer des capteurs d'image dans les téléphones mobiles. Ceux-ci laissaient entrevoir des questions de recherche dans les collections que ces capteurs génèreraient. **Le chapitre 3 présente des travaux concernant les données multimédia personnelles** que j'y ai réalisés, avec quelques prolongements menés ultérieurement au LINA.
- depuis 2000, je suis maître de conférences à l'école polytechnique de l'université de Nantes, dans le département informatique côté enseignement et au Laboratoire d'Informatique Nantes-Atlantique pour le volet recherche. Je suis également membre de l'équipe-projet INRIA Atlas, rattaché à l'IRISA. Lancé en 2004, ce projet traite de la gestion de données distribuées, et notamment de données multimédia. **Mon second axe de travail, décrit dans le chapitre 4, concerne les données multimédia dans le contexte des bases de données et les systèmes distribués**. Cette orientation tient à la fois à un contexte d'équipe et d'opportunités importantes de recherche entre ces domaines, grâce à leurs maturités respectives et à des perspectives applicatives motivantes.

Les conclusions et perspectives sont réparties en fin des chapitres 3 et 4, pour chacun de thèmes développés dans ces chapitres.

Quand le texte se réfère à nos publications, elles sont précisées en pied de page ; les autres références renvoient vers la section bibliographique page 112.

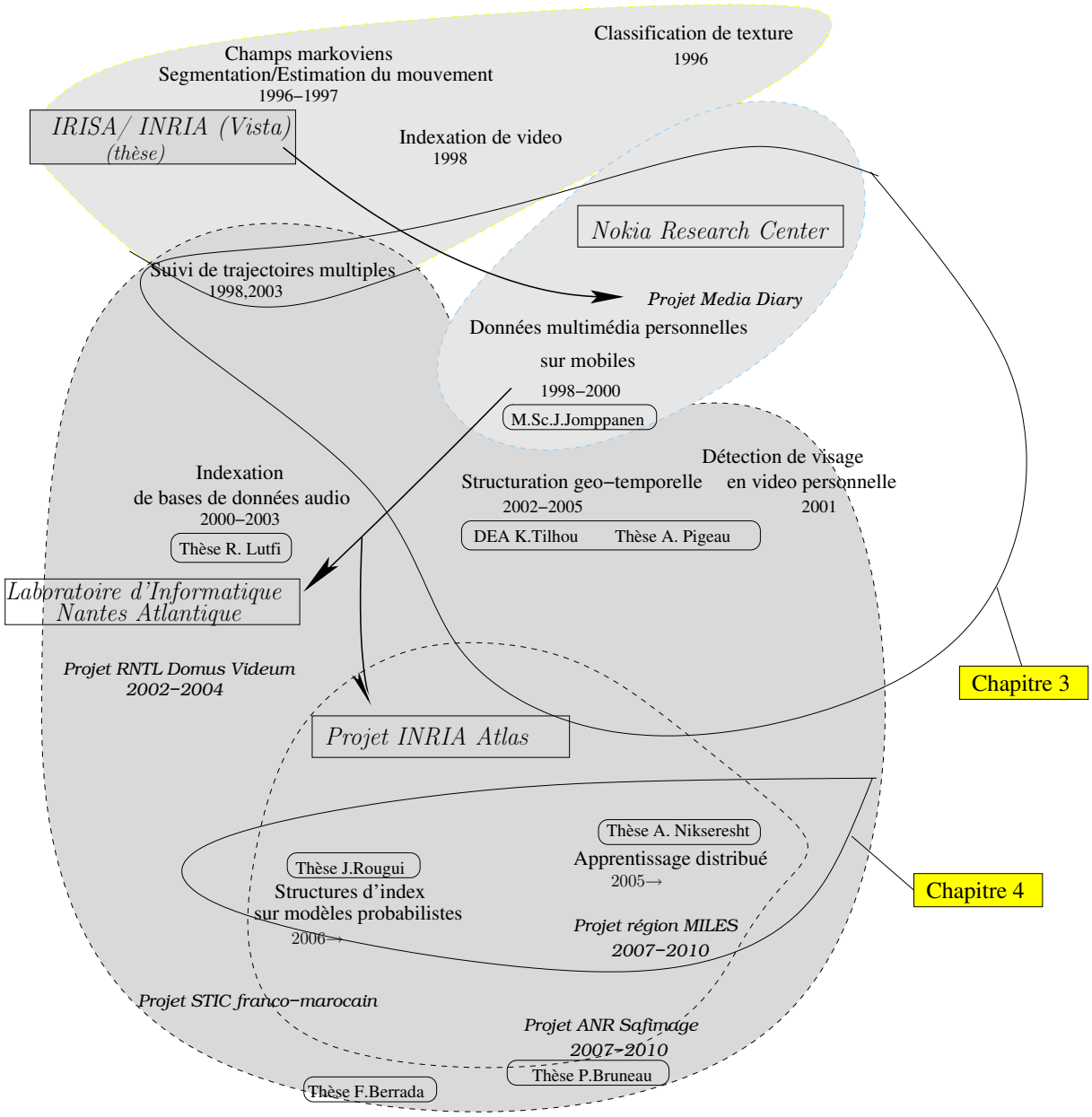


FIG. 1.1 – La figure cartographie, dans le temps, mon activité de recherche et son contexte ; la lecture des thèmes de recherche du haut vers le bas correspond à peu près à un ordre chronologique. Les quatre zones correspondent aux équipes de recherche dont j'ai fait partie.

2

Mélanges de lois

Ce chapitre présente les mélanges de lois, outil très utilisé, très polyvalent et riche de variantes, pour la résolution de problèmes de classification de données (McLachlan & Peel 2000). En particulier, ils s'appliquent souvent de manière commode et performante sur des vecteurs d'attributs issus de données audiovisuelles : les mélanges se retrouvent en reconnaissance du locuteur (Reynolds 1995), où ils caractérisent souvent la distribution d'attributs cepstraux ; en image, où ils offrent une description plus parcimonieuse que des histogrammes globaux de couleur (Hammoud & Mohr 2000, Vasconcelos & Lippman 1998), ou peuvent modéliser la distribution de probabilité de la couleur d'une classe d'intérêt ; en analyse de video, pour caractériser la distribution d'attributs temporels ou spatio-temporels, permettant de reconnaître des événements par leur dynamique (Fablet, Bouthemy & Perez 2001). Au-delà de la variété des données et problèmes applicatifs où ils ont été appliqués avec succès, la structure de mélange peut s'employer de manière plus élaborée que directement sur des vecteurs d'attributs, par exemple sur une analyse en composantes principales probabilisée (Tipping & Bishop 1999). Ils sont utilisés tantôt à fin de modélisation (supervisée) de la densité de probabilité d'une seule classe, qu'ils peuvent - en principe - représenter avec une précision arbitraire, tantôt pour en discriminer plusieurs, de manière non supervisée. Les travaux présentés dans ce document relèvent pour partie de la caractérisation d'une classe, pour partie de la discrimination entre classes. Nombre de modèles probabilistes sont variantes des mélanges de lois (dont les modèles de Markov cachés). Les lois élementaires, que le mélange combine, peuvent prendre diverses formes, même si la loi gaussienne est largement la plus utilisée, son aptitude à la modélisation de concentrations étant grande et sans fort a priori, et les calculs généralement plus simples qu'avec d'autres lois. Enfin, l'estimation des quantités inconnues et intéressantes qui paramétrisent ces modèles peut être conduite selon divers critères (maximum de vraisemblance, estimation bayesienne). En résumé, les mélanges de loi sont un carrefour en ce qui concerne la classification de données multimédia.

Dans ce document, nous évoquerons essentiellement des techniques de classification à base de modèles probabilistes générateurs, c.a.d. qui cherchent à caractériser chaque classe séparement, et où les données dont supposées être des réalisations liées au modèle probabiliste

qu'on tente de construire. Par contraste, les approches discriminantes préfèrent construire directement les frontières de décision entre classes. Sans comparer ici les propriétés des deux démarches, indiquons que des travaux récents cherchent à combiner leurs qualités : la possibilité d'opérer complètement de manière probabiliste sur des techniques à noyaux discriminantes (Tipping 2001), avec ses retombées directes pour les applications en image (Williams, Blake & Cipolla 2005) ; la possibilité de bien gérer les classes multiples (éventuellement, apparaissant en cours d'apprentissage) par des machines à vecteurs support (Hsu & Lin 2002). Enfin, des travaux ont été consacrés à l'examen de principe de l'hybridation des techniques génératrices et discriminantes (Lasserre, Bishop & Minka 2006).

2.1 Forme du modèle

Nous définissons ici des aspects essentiels de la forme du modèle [1], tandis que la section suivante présentera l'estimation des quantités qui le définissent complètement. Soit X l'ensemble des données, supposées tirées d'une distribution de probabilité $p(X)$. $p(X)$ prend la forme d'une combinaison linéaire de lois élémentaires $p_k(X)$, où $p_k(X)$ est paramétrée par θ_k :

$$p(X) \;=\; \sum_{k=1}^{K} \pi_k \; p_k(X|\theta_k) \tag{2.1}$$

On pourrait clore là cette description, mais elle n'amène pas à un procédé pratique d'estimation des paramètres $\{\pi_1, \ldots, \pi_K, \theta_1, \ldots, \theta_K\}$.

On lui préfère donc la formulation suivante, utile à plusieurs des principales méthodes d'estimation des paramètres. On introduit un vecteur aléatoire $Z = \{Z_1, \ldots, Z_K\}$, où $Z_k \in \{0, 1\}, k = 1, \ldots, K$ avec $p((Z_k = 1) = \pi_k)$. Z permet de construire la loi jointe $p(X, Z)$, qui se factorise favorablement : $p(X, Z) = p(Z)p(X|Z)$, où les distributions $p(Z)$ et $p(X|Z)$ prennent chacune une forme simple et se partagent les paramètres (resp. les π_k et les θ_k). La section suivante va montrer que cette reformulation rend plus aisée l'estimation des paramètres, qu'en procédant directement avec $p(X)$.

2.2 Estimation par algorithmes CM et EM

Parce qu'ils sont centraux à nombre de nos travaux, nous présentons ici les algorithmes CM (*Classification-Maximization*) et EM (*Expectation-Maximization*). On se familiarise souvent avec ces techniques par les mélanges de gaussiennes, dont l'estimation des paramètres prend la forme d'un algorithme bien connu, calculant, en alternance, paramètres de chaque modèle et affectations modèles-données. On retrouve cet algorithme d'ailleurs hors du cadre probabiliste.

Plutôt que rappeler les équations finales, proches de la mise en oeuvre, nous présentons ici une justification en amont. Visant la recherche de θ et de Z, nous présentons deux démarches

[1] L'objectif étant ici de fournir l'idée générale, on abandonne la distinction de notations entre variable aléatoire et réalisation, au profit d'une notation plus compacte.

alternatives, correspondant à des critères d'optimalité distincts :

• Une première possibilité est d'**optimiser la loi jointe** $p(X, Z|\theta)$. Cette optimisation peut être menée par l'algorithme nommé ici Classification-Maximization (CM), décrit dans l'Algorithme 1 ci-dessous. Il ressemble à un algorithme k-means, plongé dans un cadre probabiliste, avec plus de liberté que le k-means quant à la forme des lois (classiquement des gaussiennes avec diverses contraintes possibles sur les matrices de covariance, ou une autre forme). Une étude complète de ses propriétés peut être trouvée dans (Biernacki 1998).

Algorithme 1 Algorithme Classification-Maximization, optimisant la loi jointe $p(X, Z|\theta)$

$i = 0$, fournir $\widehat{Z^0}$ (affectations initiales des données aux modèles)
REPETER

$$1.\ \widehat{\theta^i} = \arg\max_{\theta}\ p(X|\widehat{Z}^i, \theta)$$

$$2.\ \widehat{Z}^{i+1} = \arg\max_{Z}\ p(Z|X, \widehat{\theta^i})$$

$i \leftarrow i + 1$
JUSQU'A convergence (atteinte en un nombre fini d'étapes)

A l'étape 1, parce qu'on suppose une association modèle-données $\widehat{Z}^i$, la difficulté lié à l'aspect "mélange" disparaît et, dans le cas fréquent où $p(X|\widehat{Z}^i, \theta)$ est pris dans la famille exponentielle - qui inclut la loi gaussienne - l'estimation de $\widehat{\theta}$ au maximum de vraisemblance est classique et bénéficie d'une expression analytique.

Une caractéristique importante de l'Algorithme 1, étape 1, est que $\widehat{\theta^i}$ est estimé dans le contexte où une seule hypothèse d'affectations données-modèles $\widehat{Z}^i$ est retenue. Il s'agit là d'une différence centrale avec l'algorithme EM présenté ci-dessous.

• Une second possibilité est d'**optimiser la loi marginale** $p(X|\theta)$, l'inférence sur Z sera alors un sous-produit appréciable de l'optimisation de $p(X|\theta)$. Pratiquement, on optimisera plutôt $\log p(X|\theta)$.

Dans de nombreux cas, au nombre desquels ceux qui nous intéressent dans ce document, l'optimisation directe de $p(X|\theta)$ est généralement malaisée et l'introduction de Z est un mécanisme permettant de mener à bien l'estimation. En quelques mots, le principe consiste à construire une succession d'approximations de $p(X|\theta)$ qui pourront, elles, être facilement optimisées parce qu'elles s'appuient sur $p(X, Z|\theta)$. Ces approximations seront de qualité croissante, fournissant une séquence de valeurs estimées pour θ de qualité croissante.

La loi marginale $p(X|\theta)$ et la loi jointe $p(X, Z|\theta)$ sont liées de deux façons, par des propriétés élémentaires suivantes :

$$\log\ p(X|\theta) \quad = \quad \log \sum_{Z} p(X, Z|\theta) \tag{2.2}$$

$$\log\ p(X|\theta) \quad = \quad \log\ p(X, Z|\theta)\ -\ \log\ p(Z|X, \theta) \tag{2.3}$$

L'expression (2.3) lie les trois quantités "clés" qui vont nous intéresser dans toute la discussion qui suit. Il y apparaît que ce qui sépare les deux distributions précédentes est la distribution des variables cachées, $\log\ p(Z|X,\theta)$, que l'on ne connaît malheureusement pas. A sa place, introduisons alors une distribution $q(Z)$.

$q(Z)$ va nous servir à réaliser une décomposition de $\log p(X|\theta)$ comme suit :

$$\begin{aligned}
\log p(X|\theta) &= \log p(X|\theta)\ .\ \sum_Z q(Z) \quad \text{car} \sum_Z q(Z) = 1 && (2.4)\\
&= \sum_Z q(Z)\ \log p(X|\theta) && (2.5)\\
&= \sum_Z q(Z)[\log\ p(X,Z|\theta)\ -\ \log\ p(Z|X,\theta)] \text{ via } (2.3) && (2.6)\\
&= \sum_Z q(Z)\ [\log\ p(X,Z|\theta) - \log q(Z) - \log\ p(Z|X,\theta) + \log q(Z)] && (2.7)\\
&= \underbrace{\sum_Z q(Z) \log \frac{p(X,Z|\theta)}{q(Z)}}_{\text{noté } \mathcal{L}(q,\theta)} + \underbrace{\sum_Z q(Z) \log \frac{p(Z|X,\theta)}{q(Z)}}_{\mathrm{KL}(q(Z)\ \|\ p(Z|X,\theta))} && (2.8)
\end{aligned}$$

La décomposition (2.8) est illustrée sur la figure 2.1(a). Les remarques suivantes l'éclairent :

1. Dans le second terme, il est intéressant de voir $q(Z)$ comme une approximation de la distribution a posteriori des variables cachées $p(Z|X,\theta)$.
2. Comme $\mathrm{KL}(q(Z) \parallel p(Z|X,\theta)) \geq 0,\ \forall q$, $\mathcal{L}(q,\theta)$ est toujours un minorant de $\log\ p(X|\theta)$.
3. Ce minorant $\mathcal{L}(q,\theta)$ est d'autant plus proche de $p(X|\theta)$ que l'approximation de $\log\ p(Z|X,\theta)$ par $q(Z)$ est bonne.
4. La démarche ci-dessus était d'abord motivée par ce que le minorant $\mathcal{L}(q,\theta)$ est, lui, facile à optimiser relativement à θ (pour $q(Z)$ donné) :

$$\begin{aligned}
\arg\max_\theta \mathcal{L}(q,\theta) &= \arg\max_\theta \sum_Z q(Z)\ \log p(X,Z|\theta)\ - \underbrace{\sum_Z q(Z)\ \log q(Z)}_{\text{entropie de la partition}} && (2.9)\\
&= \arg\max_\theta \underbrace{\sum_Z q(Z)\ \log p(X,Z|\theta)}_{\mathbb{E}_{q(Z)}\{\log p(X,Z|\theta)\}} && (2.10)
\end{aligned}$$

Pour tenter de mettre en relation cette démarche avec l'approche "loi jointe" décrit précédemment, observons que, dans (2.10), le vecteur θ recherché va être optimisé non pas en considérant pour *une seule* configuration Z, comme c'était le cas dans l'algorithme CM, mais ici *toutes* les hypothèses d'associations sont envisagées simultanément, pondérées de leurs probabilitées (estimées) respectives. En d'autres termes, on préfère *marginaliser* par rapport à Z plutôt qu'*optimiser* par rapport à Z.

Nous pouvons maintenant optimiser $\log\ p(X|\theta)$ en tirant parti de la décomposition (2.8). On va, en fait, optimiser $\mathcal{L}(q, \theta)$ relativement à ses deux paramètres, en alternance (voir Algorithme 2.2 et Figure 2.1).

Remarquons qu'on a montré l'amélioration monotone de $p(X|\theta)$ au fur et à mesure des itérations.

La présentation faite ci-dessus de l'algorithme EM suit la démarche de (Neal & Hinton 1998) en laissant de côté l'analogie avec la physique. Elle n'est pas la seule possible, mais elle est la plus instructive que nous ayions rencontrée. En effet, si on voulait réaliser une estimation bayesienne du mélange, les mêmes idées peuvent être reprises assez directement pour l'approximation variationnelle de $p(\theta, Z|X)$. Par ailleurs, comme présenté (Neal & Hinton 1998), elle permet de démontrer la validité de diverses extensions intéressantes.

2.3 Critère bayesien pour la détermination du nombre de composantes

Alors que le scénario précédent suppose connu (ou, de manière plus réaliste, fixé) le nombre de composantes du mélange, nous évoquons ici la détermination de ce nombre de composantes. Si on considère les techniques de clustering plus largement que dans le cadre probabiliste choisi ici, la détermination du nombre de clusters est un problème central, à la fois parce qu'elle est importante pour les applications et parce qu'elle est l'objet d'une abondante littérature, par diverses voies (bayesienne, théorie de l'information, autres critères statistiques ou issus de la logique floue), qui se rejoignent parfois. Dans nos travaux, nous avons privilégié l'approche bayesienne à cette résolution, elle-même l'objet de nombreuses recherches. Nous en résumons le principe.

Dans la section précédente, nous avons cherché à optimiser $p(X|\theta)$ au maximum de vraisemblance, par l'algorithme EM. Ce procédé ne résoud pas plusieurs besoins importants :

1 on souhaite éviter des configurations dégénérées où la vraisemblance est très élevée du fait de la variance estimée trop faible d'une ou plusieurs composantes, due à un effectif trop faible ;

2 le nombre de composantes du mélange ne saurait être déterminé par le critère du maximum de vraisemblance, car il sera d'autant mieux satisfait que le nombre de composantes augmente.

Dans la recherche du modèle $\mathcal{M}$ de complexité adéquate, la remarque 2 ci-dessus nous suggère d'optimiser $p(X|\mathcal{M})$ plutôt que $p(X|\mathcal{M}, \theta)$. Dans le cas de l'optimisation de $p(X|\theta)$ par l'algorithme EM, nous avions préféré traiter Z par marginalisation que par optimisation. La même idée est employée ici quant à la manière de prendre en compte θ dans $p(X|\mathcal{M})$:

$$p(X|\mathcal{M}) = \int_\theta p(X|\theta, \mathcal{M}) p(\theta|\mathcal{M})\ d\theta \tag{2.11}$$

où $p(\theta|\mathcal{M})$ est une distribution a priori sur θ.

$i = 0$, fournir $\widehat{Z^0}$
REPETER
Etape E : Cette étape est illustrée sur la figure 2.1(b). Supposons que l'on dispose d'une valeur θ^i. Soit $q(Z)$ la distribution à optimiser à cette étape. Dans (2.8), $\log p(X|\theta)$ ne dépend pas de $q(Z)$, donc cherchant à réduire $\mathrm{KL}(q(Z) \parallel p(Z|X,\theta))$, on augmente $\mathcal{L}(q,\theta)$. En choisissant $q(Z) = p(Z|X,\theta)$, on peut même annuler $\mathrm{KL}(q(Z) \parallel p(Z|X,\theta))$. La distribution $p(Z|X,\theta)$ est inconnue, mais on peut l'approximer, si on dispose d'une valeur approximative de θ.
Etape M : Cette étape est illustrée sur la figure 2.1(c). On garde cette fois $q(Z)$ constant et on optimise $\mathcal{L}(q,\theta)$ relativement à θ. On aura une expression analytique sympa pour cela. Ici référencer l'expression au-dessus. Comme $\mathrm{KL}(\, p(Z|X,\theta) \parallel q(Z))$ est inchangé dans cette opération, le gain obtenu sur $\mathcal{L}(q,\theta)$ sera intégralement répercuté sur $p(X|\theta)$.
$i \leftarrow i + 1$
JUSQU'A convergence

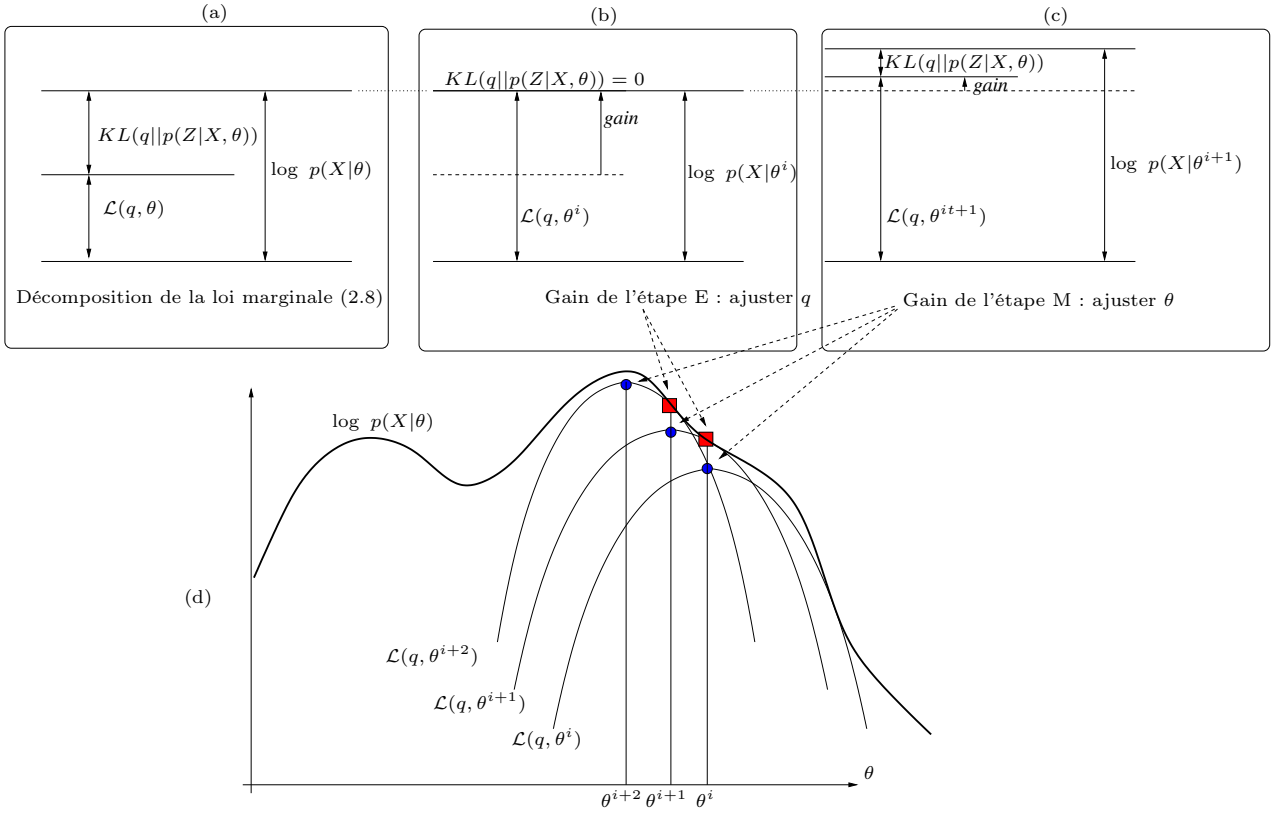


FIG. 2.1 – Principe de l'algorithme EM : (haut) Algorithme ; (bas) (a) la décomposition de $p(X|\theta)$ permettant de faire apparaître la vraisemblance jointe $p(X,Z|\theta)$ et la distribution a posteriori des variables cachées $p(Z|X,\theta)$. (b) L'étape E (c) L'étape M (d) La séquence d'approximations de $p(X|\theta)$ par $\mathcal{L}(q,\theta)^i, \mathcal{L}(q,\theta)^{i+1}, \ldots$

.

Une propriété essentielle de la marginalisation par rapport à θ, dans ce contexte, est que l'espace où intégration est réalisée voit sa dimension augmenter avec le nombre de composantes. Il en résulte un mécanisme de pénalisation automatique de la complexité de modèle, amenant à un compromis en principe révélateur du "vrai" processus générateur des données.

Nous omettons ici la discussion sur le choix des distributions a priori : nos travaux ont soit considéré des distributions a priori conjuguées habituelles pour les mélanges de gaussiennes, soit utilisé des approximations indépendantes de telles lois a priori. La démarche d'approximation la plus commune suppose que la densité de $p(X|\theta)$ est concentrée autour du mode θ_{MAP}, et qu'elle suit à peu près une loi gaussienne. L'évaluation du hessien $-\nabla\nabla \log\ p(X|\theta_{MAP})p(\theta_{MAP})$ permet alors de calculer ce volume (approximation de Laplace). Des hypothèses supplémentaires mènent à une expression encore plus simple : le critère d'information bayesien (BIC) :

$$\log p(X|\mathcal{M}) \ \approx \log p(X|\theta_{MV}) - \frac{1}{2} m \log n \tag{2.12}$$

où θ_{MV} est un estimé de θ au maximum de vraisemblance (classiquement fourni par l'algorithme EM), m est le nombre de paramètres indépendants dans le modèle et n le nombre de données sur lequel l'estimation a été réalisée.

Ces approximations, surtout le critère BIC, ont l'avantage d'être d'un emploi aisé et de fournir une approximation généralement fiable quand on estime les paramètres du mélange sur d'assez grands volumes de données. Dans des cas d'effectifs insuffisants que nous avons rencontrés pour divers problèmes, il a fallu remédier à son imprécision. Un second inconvénient est que ce critère permet d'évaluer la qualité d'un modèle, mais ne mène pas directement à un algorithme performant pour explorer un ensemble de modèles de complexités diverses, au delà d'une recherche exhaustive de l'ensemble des modèles. Enfin, le problème évoqué dans la remarque 1 plus haut n'est pas résolue, puisque θ est estimé au maximum de vraisemblance. Nous évoquons dans la section 4.3 l'approximation variationnelle de la distribution a posteriori $p(\theta, Z|X)$ qui est à même de résoudre ces difficultés par un algorithme itératif analogue à l'algorithme EM.

2.4 Conclusion

La recherche de structure dans des données multimédia requiert souvent de discriminer les sous-populations présentes, et d'en décrire les caractéristiques. Ce chapitre a présenté quelques points qui nous paraissent importants concernant les mélanges de lois de probabilité et l'estimation des paramètres et partition associée. En effet, les travaux décrits dans le chapitre suivant font une large utilisation, avec des extensions et variantes destinées à résoudre des problèmes applicatifs variés en structuration de données multimédia. Dans les chapitres suivants, nous utilisons ce cadre tantôt pour estimer la densité d'une classe de façon semi-paramétrique, tantôt pour partitionner un ensemble de données en classes.

3

Structuration des données multimédia personnelles

Ce chapitre présente un ensemble de travaux concernant la recherche d'information dans les données multimédia personnelles "grand public" (collections d'images, en particulier). Cette question est contemporaine de la massification des capteurs d'images numériques. Dans la mesure où il s'agit souvent de structurer des collections de données, les mélanges de loi ont été un outil central dans les techniques que nous avons proposées.

Dans un premier temps, je présente un travail réalisé dans l'industrie, à l'origine de mon intérêt pour ce thème. Ma revendication ici concerne la pertinence de choix d'orientation et d'impulsions, dans un contexte de "défrichage", et dans la gestion du projet pendant 20 mois, plutôt qu'une contribution purement scientifique. En conséquence, la description est plus narrative qu'il est d'usage. Dans un second temps, ce thème a été poursuivi dans le cadre académique, nous le décrivons alors sous l'angle plus scientifique.

3.1 Contexte

Je suis arrivé en décembre 1998 dans le groupe de recherche de codage video, dirigé par Marta Karczewicz, à Nokia Research Center, Tampere. J'ai été recruté pour lancer un projet "autour de MPEG-7" (sic). La motivation pour l'image et la video étaient alors concentrée sur la compression et la transmission robuste, et, côté matériel, les dispositifs d'acquisition et d'affichage d'image. Simultanément, l'entité de production Nokia Mobile Phones/Oulu explorait l'informatique vestimentaire et ubiquitaire, par des scénarii d'utilisation très futuristes. Enfin, un projet Tekes (sorte d'ANR finlandais) venait d'être initié entre le groupe que j'intégrais et l'Université de Technologie de Tampere, sur la recherche d'image par le contenu ; les travaux avaient été orientés vers une recherche basée sur les formes, couleurs, textures, histogrammes globaux, alors également en vogue dans MPEG-7. Le mode privilégié était la requête par l'exemple, avec présentation des résultats dans l'ordre de similarité.

3.2 Orientations générales

Dans ce contexte, manquant de cohérence, il m'a été confié de définir des axes de travail, puis de mener un projet R&D dans le domaine.

Il m'a semblé que :

- le "wearable computing", caractérisé par les terminaux et capteurs futuristes, était un concept intéressant mais avait un avenir assez incertain, alors que des tâches à plus court-terme, dont l'utilité était plus probable, restaient à accomplir.
- le trio forme/couleur/texture et la forme de requête par similarité n'apparaissaient pas directement utiles à la marge majorité des besoins grand public, que nous visions.

En conséquence, j'ai choisi d'orienter le travail comme suit :

- permettre de trouver des informations selon des critères plus proches des souhaits vraisemblables de l'utilisateur : personnes présentes, temps et lieu de la prise de vue
- étant donnés les critères choisis, construire, par analyse automatique, une structure sur la collection permettant la navigation selon ces critères, plutôt que des requêtes.
- j'ai établi les contacts avec les équipes préparant 1) les capteurs photo intégrés et le traitement d'image pour la restitution, qui devaient être intégrés dans les téléphones produits deux ans plus tard, et choisi de travailler sur les collections d'images (fixes) que l'on pourrait acquérir avec ces terminaux 2) les systèmes de géolocalisation qui devaient être intégrés dans les terminaux

Ces orientations paraissent, a posteriori, assez évidentes. Malgré tout, en 1999, dans un contexte où la recherche d'image par comparaison était reine côté académique et où l'industrie avait encore peu proposé, ça n'était pas le sens de l'histoire. Les premières publications en la matière datent du début des années 2000 : concernant l'analyse des besoins (Rodden 2003), qui confirmait nos intuitions, et des propositions techniques basées une classification non-supervisée des estampilles temporelles (Gargi, Deng & Tretter 2002, Graham, Garcia-Molina, Paepcke & Winograd 2002, Platt & M. Czerwinski 2002) puis de géolocalisation (Ashbrook & Starner 2002). Le nombre de publications et réunions sur ce thème (ACM Carpe Workshop) a augmenté régulièrement depuis ces années.

L'équipe consacrée à ce projet est passée de 3 à 5 personnes. Le projet se poursuit à ce jour et compte 16 personnes. Il est passé en 2002 du centre de recherche aux unités plus proches des produits. Un aboutissement direct est le logiciel commercial Nokia Lifeblog.

3.3 Travaux concernant l'image

3.3.1 Détection de visage

Avec une collègue (Doina Petrescu) titulaire d'un doctorat en traitement d'image, nous avons examiné la question de la détection de visage dans des images acquises avec les futurs téléphones, pour une application grand public. Ceci fixait les particularités suivantes :

- le même capteur optique acquiert toutes les images à traiter, ce qui permet de calibrer la techique de détection sur les caractéristiques de couleur du capteur,
- une résolution assez médiocre, présence fréquente de flou de mouvement, pas de flash,
- la variabilité de taille des visages, leur pose très variable dans les collections "prototypes" rendent délicates les solutions basées sur l'apprentissage de l'organisation spatiale de l'intensité (Eigenfaces et améliorations (Rowley, Baluja & Kanade 1998)).
- un coût calculatoire modeste, le traitement devant être réalisé sur le terminal.

Nous avons proposé une technique de localisation de visage alliant un apprentissage statistique supervisé dans un espace teinte-saturation (ignorer la luminance permet une certaine invariance à l'éclairage).On suppose les distributions $p((\text{teinte,saturation})|\text{couleur de peau})$ et $p((\text{teinte,saturation})|\overline{\text{couleur de peau}})$ peuvent être décrites par des mélanges de de gaussiennes, dont on estime les paramètres par l'algorithme EM. En fait, pour s'adapter aux conditions globales d'illumination, on construit plusieurs lois de probabilité conditionnelle. Le choix de loi s'effectue à l'examen global de l'image où les visages doivent être détectés. Un traitement de morphologie mathématique permet ensuite d'améliorer la fiabilité des zones détectées, par la prise en compte du contexte spatial. Ce travail a été publié dans un congrès européen [2]

3.3.2 Structuration temporelle basée sur la présence de visage

Une demande applicative forte consiste à identifier les épisodes où l'utilisateur a interagi avec d'autres personnes. Une motivation applicative voisine qui suscite régulièrement des publications est l'aide à la recherche d'information dans les réunions filmées (Wellner, Flynn & Guillemot 2004). Partant d'image acquises, à quelques secondes d'intervalle, d'une minicaméra dont le champ de vue est similaire à celui de l'utilisateur, le principe a été d'exploiter la détection de visage. L'originalité du besoin est là qu'on n'a pas besoin de la localisation du visage dans l'image, ni d'ailleurs de localisation précise dans le temps. Il ne s'agit que de retrouver grossièrement, dans le temps, les segments temporels où la présence de visage est probable et durable. Dans ce travail, l'observation construite veut refléter la probabilité d'une image de contenir au moins un visage. La segmentation temporelle est formulée comme un problème d'étiquetage en 2 états : présence/absence d'au moins visage dans l'image. L'état dépend bien sûr de l'observation, de manière probabiliste, mais comme les images de caméra portée sont de faible qualité (cf. section 3.3) et l'observation assez peu fiable, et parce qu'on souhaite construire une structure compacte, on régularise temporellement l'étiquetage (le modèle est alors une chaîne de Markov cachée non orientée). La programmation dynamique permet de trouver la séquence d'états la plus probable, à faible coût et, si besoin, de manière incrémentale. Ce travail a été présenté au congrès IEEE ICIP'2001[3]

[2] D. Petrescu, M. Gelgon, *Face detection from complex scenes in color images*, in Proceedings of EURASIP European Signal Processing Conference (EUSIPCO'2000), Pages 933-936, Tampere, Finland, Septembre 2000.

[3] M. Gelgon, *Using face detection for browsing personal slow video in a small terminal and worn camera context*, in IEEE. Int. Conf. on Image Processing (ICIP'2001), Pages 1062-1065, Thessaloniki, Greece, Septembre 2001.

3.4 Structuration de séquence de mesures de géolocalisation

La disponibilité, à terme, d'une information de géolocalisation dans les mobiles (capteur embarqué ou mesure par le réseau), permet de conserver en mémoire la trajectoire de l'utilisateur. La recherche de structure dans cette trajectoire doit aboutir à un découpage geo-temporel où se retrouvent à la fois les lieux et les segments temporels significatifs. Leur détermination permet d'alimenter un système de recherche dans les documents personnels bénéficiant, pour naviguer, d'une structure tentant de refléter l'activité passée de l'utilisateur, dans la mesure où elle se concrétise par des mesures géo-temporelles.

3.4.1 Travaux initiaux, Nokia Research Center

Si l'utilisation de la position géographique est, de prime abord, séduisante, des obstacles techniques demeuraient :

- une incertitude sur le système qui, en pratique, fournirait les coordonnées (GPS, combinaison GSM-GPS ou E-OTD triangulant des stations de base) ;
- le fait que les coordonnées (latitude,longitude) à l'état "brut" ne sont pas commodes ni pour l'interrogation, ni pour la visualisation, mais avec des doutes forts sur la prochaine disponibilité d'un système d'information géographique externe permettant de fournir des étiquettes symboliques
- l'indisponibilité temporaire des signaux GPS, en fonction des conditions de propagation radio.

Sur ce point, nous avons proposé, dans le cadre du master de Jarmo Jomppanen (1999-2000) :

- une technique d'estimation de la trajectoire lors des périodes -très fréquentes- d'indisponibilité des signaux GPS, au moyen de filtrage de Kalman avec lissage ;
- une technique pour identifier les "lieux importants" de l'utilisateur, révélés par des concentrations de données (on suppose acquérir et stocker la position courant régulièrement). La solution technique est assez classique, en ce qu'un algorithme avec une classification de type CM, avec matrices de covariances sphériques, est employée pour identifier les zones importantes, mais elle a été l'objet d'un volet expérimental conséquent (travail en lien avec l'entité Nokia intégrant les GPS dans les terminaux, constitution de corpus de données, évaluation des résultats) et, à l'époque, original. Ce travail n'a malheureusement été déposé que sous la forme d'un rapport interne d'invention Nokia.

Nous décrivons ci-dessous la poursuite de ce thème au LINA.

3.4.2 Structuration de séquence par mélange de lois

Nous avons d'abord travaillé sur une séquence de données, où ces observations sont supposées acquises régulièrement (à intervalle de quelques secondes). Nous avons cherché à partitionner la trajectoire comme une séquence de portions de trajectoire dans l'espace tri-dimensionnel $(\text{latitude}, \text{longitude}, \text{temps})$, chaque portion devant être décrite par un polynôme, le nombre

de portions devant également être déterminé. Nous avons formulé ce problème comme un mélange de lois, résolu par l'algorithme EM, où l'étape 'M' consiste en une regression polynômiale. Parce que l'étape 'E' ne prend pas en compte l'aspect séquentiel des données (c'est ce qui confère à l'algorithme son faible coût), une initialisation et une stratégie de recherche spécifiques sont proposées, pour déterminer le nombre de segments, en visant un critère BIC. Ce travail a été présenté au congrès IEEE ICME'2002, joint à la fin de ce chapitre. [4]

Ce thème s'est poursuivi dans le cadre de la thèse d'Antoine Pigeau (2002-2005). Nous avons conservé le cadre les mélanges de lois, mais n'avons plus opéré par regression. Le travail s'est enrichi sur des points suivants :

- **incrémentalité** de la structuration : les données ayant vocation à être acquises et consultées de manière "entrelacée", il s'agit de traiter un flux de données au fur et à mesure de sa génération. Si l'algorithme EM, par son caractère d'optimisation locale, se prête naturellement à un mécanisme de mise à jour des paramètres avec l'arrivée de nouvelles données, sa simple application rend notre problème est rapidement victime de minima locaux. Des études ont fourni des stratégies d'initialisation (Biernacki, Celeux & Govaert 2003), mais dans le cas incrémental, nous avons préféré opter pour une stratégie "semi-locale" d'exploration [5], permettant un bon compromis entre l'aptitude à remettre en question les groupes de données, la cohérence temporelle des partitions et le coût calculatoire impliqué.
- **robustesse face aux erreurs de modèle**
  Dans notre contexte applicatif, le processus générateur des données est, malheureusement, assez loin de respecter l'hypothèse de gaussianité. En fonction de la configuration des données, la qualité de la partition des données Z estimée peut en souffrir peu ou beaucoup. Nous avons mis en oeuvre et évalué (thèse d'Antoine Pigeau) un mécanisme permettant de mieux résister à cette difficulté. Il peut s'expliquer comme suit :
  Un ensemble de données correspondant intuitivement à un groupe, mais mal décrit par une loi gaussienne aura tendance a être mieux décrit par plusieurs gaussiennes, même avec un critère bayesien, qui n'en introduit qu'avec parcimonie. Comme cet ensemble présente, par nature, une densité assez forte, il va généralement présenter un sous-ensemble pour lequel les données ne sont pas nettement issues d'une seule des gaussiennes décrivant l'ensemble, c.a.d. que, dans l'algorithme EM, $\widehat{q(Z)}$ obtenu à convergence présente une entropie élevée. En modifiant le critère d'optimalité pour favoriser l'obtention de groupes nettement séparés, on peut donc améliorer la robustesse de la classification à une erreur de modèle. Cette discussion nous ramène à l'expression (2.9), rappelée ici :

$$\arg\max_{\theta} \mathcal{L}(q,\theta) \;=\; \arg\max_{\theta} \underbrace{\sum_{Z} q(Z)\ \log p(X,Z|\theta)}_{\mathbb{E}_{q(Z)}\{\log p(X,Z|\theta)\}} \;-\; \underbrace{\sum_{Z} q(Z)\ \log q(Z)}_{\text{entropie de la partition}} \qquad (3.1)$$

A convergence de l'algorithme EM, où sont obtenus des paramètres $\widehat{\theta}$ et une partition

[4] M. Gelgon, K. Tilhou, *Structuring the personal multimedia collection of a mobile device user based on geolocation*, in IEEE Int. conf. on Multimedia and Expo (ICME'2002), Pages 448-451, Lausanne, Switzerland, Août 2002.

[5] A. Pigeau, M. Gelgon,*Incremental statistical geo-temporal structuring of a personal camera phone image collection*, in Proc. of Int. Conf. on Pattern Recognition (ICPR'2004), Pages 224-228, Cambridge, U.K, 2004

$\widehat{q(Z)}$, la relation (3.1), ré-écrite en (3.2), montre un lien entre les critères *loi jointe* et *loi marginale* : le critère de *vraisemblance complétée* $p(X, \widehat{Z}|\theta)$ (Biernacki, Celeux & Govaert 2000), soustrait l'entropie de la partition, à la vraisemblance obtenue pour la loi marginale. Ce procédé permet ainsi d'introduire, dans un seul critère, donc sans pondération ad hoc, deux propriétés souhaitées pour le mélange recherché : d'une part, une bonne adéquation des modèles aux données ; d'autre part, des groupes de données bien séparées - ce qui pratiquement apporte une robustesse aux erreurs de modèles. Si on peut optimiser la vraisemblance completée directement par l'algorithme CM, le travail expérimental mené lors de la thèse d'Antoine Pigeau a montré que de meilleurs résultats étaient obtenus en optimisant la loi marginale par l'algorithme EM, puis en soustrayant le terme entropique.

$$p(X, \widehat{Z}|\theta) \;=\; p(X|\theta) \;+ \widehat{q(Z)} \;\log\; \widehat{q(Z)} \tag{3.2}$$

Il serait intéressant de comparer la robustesse que nous avons atteinte avec cette que fournirait un mélange de lois de Student. Par sa queue lourde, une loi de Student permet une meilleure tolérance aux données qui s'éloignent fortement du modèle qu'une lois gaussienne. Si l'estimation de ses paramètres est plus délicate que dans le cas gaussien, de bonnes solutions existent grâces à la décomposition d'une loi de Student comme une somme infinie de gaussiennes de même espérance, mais avec un lien variance/poids régi par une loi gamma (Bishop & Svensen 2004).
L'application nous a, enfin, confronté au problème des petits échantillons, avec des conséquences sur la fiabilité de l'estimation du nombre de composantes et des matrices de covariance. Pour le second point, des résultats satisfaisants ont été obtenus par une régularisation de la matrice de covariance, augmentant son nombre de degrés de liberté avec le nombre de données jugées originaires de la composante concernée. Nous avons aussi tenter de procéder à une estimation bayesienne de cette matrice, basé sur un a priori conjugué de Wishart, avec des résultats expérimentaux similaires et un surcoût calculatoire.

– **incrémentalité et hiérarchie de partitions**
Nous avons postulé qu'il existait généralement une structure hiérarchique dans les données, dont l'identification serait très profitable pour permettre des vues et une navigation plus aisée dans des grandes collections d'images personnelles. Nous avons proposé une technique pour extraire un ensemble de partitions de complexités diverses, chaque partition optimisant localement le critère présenté à la section précédente. A la structure arborescente est associée une stratégie de mise à jour (structure du modèle et paramètres) restreinte aux branches de l'arbre où cela est jugé nécessaire. Ce travail, intégrant les aspects d'incrémentalité et de robustesse évoqués plus haut, a été présenté dans une communication[6], jointe à la fin de ce chapitre.
En parallèle, Afshin Nikseresht débutait sa thèse sur l'estimation de mélange en contexte distribué (travail décrit dans le chapitre suivant). Son travail faisait intervenir un moyen de construire à faible coût de calcul une hiérarchie de mélange de gaussiennes (l'étage supérieur de la hiérarchie étant obtenu par une sorte de k-means opérant sur les sta-

[6] A. Pigeau, M. Gelgon, *Building and tracking hierarchical partitions of image collections on mobile devices*, in ACM Multimedia conference, Pages 141-150, Singapore, Novembre 2005.

tistiques suffisantes du mélange fin, minimisant une approximation de la divergence de Kullback entre ces deux étages.) Parce que cet algorithme de regroupement de gaussiennes était incrémental, il a pu s'appliquer directement au problème traité en thèse par Antoine Pigeau [7].

Nos travaux dans le domaine ont été cités dans la revue ACM Multimedia Systems Journal, les congrès ACM Multimedia 2006 $(\times 3)$, ACM Multimedia Information Retrieval, ACM SIG Information Retrieval, IEEE Pervasive Computing, Mobile HCI, SPIE/VCIP $(\times 2)$.

3.5 Suivi probabiliste multi-trajectoires d'objets mobiles

Nous évoquons ici un travail en marge du thème applicatif de ce chapitre, mais présentant une forte connexité scientifique avec la structuration de données géo-temporelles, quant à la méthode de résolution. Il s'est agi de poursuivre une problématique liée à mes travaux de thèse, avec P.Bouthemy et J.-P. Le Cadre (IRISA, projet VISTA), concernant le suivi d'objet dans une séquence d'images. Le point-clé du scénario considéré est la présence simultanée de plusieurs objets mobiles dans la séquence. Si on sait les détecter de manière plus ou moins fiable, il existe par contre une ambiguité dans l'association, au cours du temps, des zones détectées correspondant au même objet réel (par ex., situations de croisements). Le travail a adapté, à un problème de vision par ordinateur, une technique de suivi multi-pistes élaboré dans un cadre radar/sonar (Giannopoulos, Streit & Swaszek 1997). Ici encore, on modélise le problème comme un mélange de lois, que l'on résoud par l'algorithme EM. La spécificité du travail vient de ce que :

- l'observation construite associe la position et la forme des zones mobiles, de manière probabiliste ;
- l'état (position et forme de la région) est supposé suivre une évolution markovienne. L'étape 'M' de l'algorithme réalise alors une estimation de θ selon le critère du maximum a posteriori, par lissage de Kalman sur l'ensemble de la séquence. Ceci permet au passage d'estimer les états des objets mobiles aux instants où ils n'ont pas pu être détectés.

Notre proposition, fournie à la fin de ce chapitre, a été publiée dans la revue Image and Vision Computing [8].

Une similitude existe avec le problème précédent, concernant la structuration de séquence de mesures de géolocalisation : les modèles décrivent le mouvement d'éléments au cours du temps, mouvement animés d'une forme de continuité. Cette forme de continuité est d'ailleurs ce qui va permettre de résoudre le problème de mélange, par recherche de l'association modèle-données assurant une "continuité maximale" sur la séquence, traduite dans les modèles par une adéquation "modèle continu"-données maximale, mesurée par un critère de maximum de

[7] A. Pigeau, A. Nikseresht, M. Gelgon, *Fast tracking of hierarchical partitions with approximate KL-divergence for geo-temporal organization of personal images*, in Proc. of ACM Symposium of applied computing (SAC'2007), Multimedia and Visualization track, Pages 1088-1089, Seoul, Korea, Mars 2007

[8] M. Gelgon, P. Bouthemy, J.-P. Le Cadre, *Recovering and associating the trajectories of multiple moving objects in an image sequence with a PMHT approach*, Image and Vision Computing (Elsevier), 23(1) :19-31, 2005

vraisemblance ou maximum a posteriori. La distinction majeure concerne la façon de spécifier, dans le modèle, la cohérence temporelle de chaque trajectoire. Dans le cas du suivi d'objets, cette cohérence est introduite par un modèle d'état à évolution markovienne, elle est donc spécifiée localement dans le temps. Dans le cas des données géolocalisées, chaque cluster, ou portion de trajectoire, est modélisé par une forme paramétrique (par ex. polynomiale), imposée globalement au cluster.

Le thème du suivi d'objet dans des séquences d'images a été l'occasion de rédiger un chapitre de livre [9] sur le sujet avec Riad Hammoud (Labo. de recherche de l'équipementier automobile Delphi, USA). Il s'agit d'une synthèse sur le problème du suivi temporel d'objet et les principales techniques dans l'état-de-l'art (détection automatique d'erreur de suivi, suivi par filtre particulaire). L'originalité du texte vient de qu'il est rédigé dans l'optique de la création de vidéos grands public interactives, dont la création est assistée par un opérateur humain (analyse des conséquences d'erreurs de suivi, détection automatique de ces erreurs). J'ai coordonné la rédaction de ce chapitre et en ai rédigé la majeure partie.

3.6 Autres activités (Nokia Research Center)

Le projet décrit ci-dessus, lors de sa phase dans l'industrie, a été l'objet de travaux connexes et collaborations :

- en interne au projet, participation au travail de réflexion sur la stratégie à retenir concernant la répartition, éventuellement dynamique, des données images acquises d'un téléphone mobile, supposé relié en permanence à un réseau d'opérateur. Ce travail, un temps poursuivi après mon arrivée à l'université de Nantes, a donné lieu à un brevet [10], inséré à la fin de ce chapitre.
- en externe, en tant que représentant Nokia en 1999-2000 dans le groupe MPEG-7 *Universal Multimedia Access* (avec Ericsson, EPFL, IBM, Siemens), visant à faire prendre en compte, par le comité de normalisation, les terminaux mobiles. Alors que l'esprit général était à des descripteurs d'image classiques, j'ai poussé pour faire émerger, dans la standardisation, des descripteurs de contexte géo-temporel associés aux images.
- en interne, avec le chef de projet JPEG2000, Fehmi Chebil. JPEG2000 permet le codage par région d'intérêt, mais le mécanisme pour définir une région d'intérêt restait à fournir, et privilégier les visages s'est avéré intéressant.
- avec un étudiant chinois, Kongqiao Wang, qui débutait son doctorat à l'époque sur la reconnaissance des caractères écrits chinois, sous la direction de Jari Kangas, en subsitut du clavier. Mon idée était qu'il fallait tisser des liens avec toutes les activités en reconnaissances des formes dans l'entreprise. Nous avons, ensemble, initié un stage de master [11] sur la détection de zones de texte dans les images acquises à partir de

[9] M. Gelgon, R. Hammoud, Chapter *Object tracking and matching for building object-based hyperlinks*, in Handbook of interactive video : algorithms and technology, R. Hammoud (ed.), pp 45-65, Springer Verlag, 2006.

[10] A. Myka, Yrjänainen, M. Gelgon, *Enhanced storing of personal content*, US Patent 16660/10502275, Nokia corporation, juillet 2004.

[11] P. Heinonen (Master degree, Tampere University of Technology, Finlande, 2000), Detecting text from

téléphones mobiles. L'utilité assez immédiate que j'avais envisagée était l'utilisation du capteur image comme mini-bloc-note-photocopieuse (requérant détection de zones de texte+super-résolution).

- avec un étudiant en Master de l'université de Oulu, Jyrki Hoisko, sur les perspectives en structuration automatique des données personnelles issus de capteurs "futuristes". Nos discussions ont contribué à un article [12].
- avec des spécialistes internes de l'ergonomie et de l'interaction homme-machine sur mobiles (Virpi Roto). J'ai gardé sur ce point un intérêt pour les travaux de la communauté IHM (j'ai présenté le travail réalisé au LINA (DEA de Kevin Tilhou, que j'ai encadré) au congrès MobileHCI[13].
- avec deux ingénieurs (Markku Vehviläinen, Petri Nenonen) commençant à travailler sur la super-résolution d'image : j'avais travaillé pendant ma thèse sur l'estimation robuste 2D du mouvement au moyen de modèles paramétrés, j'ai participé à leur travail bibliographique.
- Concernant la collaboration universitaire pré-existante, j'ai demandé qu'elle soit réorientée pour les derniers mois vers le regroupement automatique d'images acquises successivement et visuellement très similaires, donc justifiant d'apparaître groupées lors de la navigation dans la collection d'images.

3.7 Apprentissage de profil utilisateur

Ce projet (2002-2004), financé par le RNTL, est issu de Thomson Multimédia R&D, à Rennes, qui concevait les dispositifs de stockage numérique de données audiovisuelles, appelés à se subsituer aux magnétoscopes des particuliers. Il s'agissait, pour ce porteur de projet, d'élaborer, d'évaluer et de démontrer des fonctions enrichissant ou facilitant d'interaction de l'utilisateur "grand public" avec la base de données audiovisuelles, potentiellement très grande, qu'il peut se constituer, alors que les sources de diffusion se multiplient également, mais les alternatives en matière de voies de transmission et équipements de réception aussi.

La contribution de notre équipe a concerné l'apprentissage, par un algorithme mis en oeuvre dans le dispositif, d'un profil des goûts de l' utilisateur, permettant de sélectionner, dans des flux télévisés reçus, ceux qui apparaissent pertinents, au vu de préférences explicitées, mais aussi implicites par des visualisations antérieures de programmes. Le problème scientifique a été ici de proposer un mécanisme permettant d'évaluer l'intérêt de manière inductive (capacité de généralisation relativement à des cas connus proches dans l'ensemble d'apprentissage).

La contribution scientifique est très largement due à Guillaume Raschia, enseignant-chercheur dans la même équipe que moi, ce projet permettant de prolonger ses travaux de thèse concer-

images taken from a camera phone.

[12] *Early Experiences of Visual Memory Prosthesis for Supporting Episodic Memory*, Jyrki Hoisko, *International Journal of Human-Computer Interaction 2003*, Vol. 15, No. 2, Pages 209-230

[13] M. Gelgon, K. Tilhou, *Automated multimedia diaries of mobile device users need summarization*, in 4th Int. Symp. on Human Computer Interaction with Mobile Devices (Mobile CHI'2002), LNCS 2411, Pages 36-44, Pisa, Italy, Septembre 2002 en 2002).

nant le résumé de données par hiérarchie de représentations linguistiques floues. Cette collaboration[14] a pris la forme du stage de DEA d'Antoine Pigeau, que nous avons co-encadré, et du contrat d'ingénieur de recherche de Gaëtan Gaumer. Ma contribution à ce projet est, par contre, majeure dans son montage et sa gestion. A ce moment, l'équipe de recherche était organisée en deux sous-thèmes assez distincts "résumés de données" et "bases de données multimédia" (auquel j'étais rattaché). Ce projet a aussi été une occasion pour moi de travaux avec le sous-thème "résumés de données", sur des points apprentissage/classification qui forment, à l'heure actuelle, une colonne vertebrale de l'équipe GRIM. Ce projet a été l'occasion de travailler avec une spécialiste des évaluations et de l'ergonomie des produits STIC de Thomson Multimedia, Izabela Grasland, qui vient depuis enseigner ce domaine à Polytech'Nantes, suite à ma sollicitation.

L'examen *a posteriori* de ce projet est intéressant : depuis 2006, les 'podcasts', plateformes participatives audiovisuelles, abonnements RSS se sont fortement développées, libérant le spectateur des contraintes de grille et de linéarité de diffusion classiques, rendant à peu près le service décrit plus haut, avec toute la puissance et la liberté de l'ordinateur individuel.

Cette question d'apprentissage de profil utilisateur a été l'occasion d'une autre (brève et informelle) collaboration [15], cette fois avec Alcatel Recherche & Innovation, cette fois en vue d'une application sur mobiles.

3.8 Conclusion

Ce chapitre a présenté une synthèse de mon travail concernant la structuration de données multimédia personnelles. Il s'est inscrit, dans un premier temps, dans un projet industriel. Je l'ai poursuivi quelques années dans le monde académique, car malgré un besoin applicatif me paraissant fort, peu de propositions avaient alors été faites. Le grand nombre de propositions parues depuis semble valider l'intérêt du problème. J'ai privilégié, dans ce thème applicatif, des problèmes de classification, parce qu'ils me semblaient centraux à la fourniture d'une forte "valeur ajoutée" pour l'accès à l'information. Simultanément, cette orientation me permettait d'évoluer, depuis l'analyse d'image, vers une compétence plus polyvalente et plus à même d'interagir à terme, de manière pertinente et non par simple juxtaposition, avec les bases de données et les systèmes distribués. Le chapitre suivant tentera de montrer la continuité scientifique dans la discontinuité applicative. Faciliter l'accès aux grandes masses de données multimédia personnelles par une structuration reste malgré tout, à mon sens, une question intéressante et importante.

Nous avons choisi de nous restreindre au seul examen des données personnelles individuelles, sans solliciter de système d'information géographique. Les travaux récents dans le domaine (Kennedy, Naaman, Ahern, Nair & Rattenbury 2007) exploitent ces derniers d'une manière

[14] A. Pigeau, G. Raschia, M Gelgon, N. Mouaddib, R. Saint-Paul, *A fuzzy linguistic summarization technique for TV recommender systems*, in IEEE Int. Conf. of Fuzzy Systems (FUZZ-IEEE'2003), Pages 743-748, St-Louis, USA, Mai 2003.

[15] A. Aghasaryan, S. Betge-Bresetz, G. Raschia, M. Gelgon, User and Usage Profiling in a Multi-Platform Service Environment, 14th Workshop on Adaptivity and User Modeling in Interactive Systems (ABIS 2006), University of Hildesheim, Pages 14-16, Septembre 2006

prometteuse, puisque qu'ils sont alimentés par les annotations de manière "participative".

3.9 Sélection de publications pour ce chapitre

Les noms des jeunes chercheurs que j'ai encadrés sont soulignés.

1. A. Pigeau, M. Gelgon, *Building and tracking hierarchical partitions of image collections on mobile devices*, in ACM Multimedia conference, full paper, Pages 141-150, Singapore, Novembre 2005

2. M. Gelgon, P. Bouthemy, J.-P. Le Cadre, *Recovering and associating the trajectories of multiple moving objects in an image sequence with a PMHT approach*, Image and Vision Computing (Elsevier), 23(1) :19-31, 2005.

3. A. Myka, Yrjänainen, M. Gelgon, *Enhanced storing of personal content*, US Patent 16660/10502275, Nokia corporation, juillet 2004.

Building and tracking hierarchical geo-temporal partitions for image collection management on mobile devices

A. Pigeau and M.Gelgon
LINA FRE 2729 CNRS / INRIA ATLAS group
2, rue de la Houssiniere
BP 92208
44322 Nantes cedex 03 - France
email: $surname$@univ-nantes.fr

ABSTRACT

Usage of mobile devices (phones, digital cameras) raises the need for organizing large personal image collections. In accordance with studies on user needs, we propose a statistical criterion and an associated optimization technique, relying on geo-temporal image metadata, for building and tracking a hierarchical structure on the image collection. In a mixture model framework, particularities of the application and typical data sets are taken into account in the design of the scheme (incrementality in the optimization phase, non-Gaussianity and the ability to cope with both small and large samples in the modelling phase). Results are reported on real data sets.

1. INTRODUCTION

Through the daily use of mobile devices (phones, digital cameras), large personal image collections are currently being built by consumers. As it is essential to provide these users with solutions for retrieving pictures efficiently among usually several thousands, the corresponding research subfield of multimedia indexing is currently attracting much interest. The Nokia Lifeblog product [17] and Microsoft MyLifeBits research prototype [13] are two recent answers from industry. Particularities of the task, compared to more classical research on multimedia content-based retrieval, come from the *image meta-data* provided by the acquisition device (time, geolocation, camera settings) and the *querying/browsing criteria* preferred by users. Studies on this latter point, reported in [8, 23, 28, 29], conclude, as one would expect, that social interaction/events, time and places are the appropriate cues to trigger memories.

In this field of consumer images, some work has addressed image content-based supervised classification (e.g. distinction between indoor and outdoor [16], face-based characterisation [11] which is now a reasonnable task to implement on PDAs with recent low-cost algorithms [27]). In contrast, the present paper focuses on the sole use of temporal and geolocation meta-data attached to each picture. We assume location are coordinates provided by a GPS/E-OTD type of equipment, i.e. the data is a stream of $\{(t,(x,y)) \in \mathbb{R} \times \mathbb{R}^2\}$ elements. Throughout, although we mention the example of an image collection, since this is the major current applicative need, the proposal is not technically tied to a particular type of document. The intended contribution is a technique (criteria and algorithm) for automatically building a hierarchical organization of images, based on their time and geolocation stamps. Such a structure obviously assumes that the generative process of pictures frequently exhibits such geo-temporal clusters and often in a hierarchical fashion. In other words, the task may be viewed as the "inverse problem" of recovering of meaningful episodes of the user's life, given images provided during these episodes. The hierarchies of partitions are built incrementally, as data flows in, since the image acquisition and collection browsing phases are highly interleaved. The goal of extracting such a spatio-temporal structure is supported by the following reasons :

1. the spatial and temporal axes employed for structuring are clear and familiar to users (the diary and map metaphores) compared to e.g. browsing according to image color features. Still, no map browsing is intented here, time-oriented views can be built on a location-based structure ;
2. at least one of the two tags of the sought document is often remembered by the user [28], but he may approach his goal by iteratively switching between the temporal and spatial views, depending on how viewing intermediate images during the search trigger reorientation ;
3. browsing, rather than querying, enables the user to navigate into a personal multimedia diary/photo album without having a particular target-picture in mind (as a passtime or to get an overview);
4. browsing along these axes is feasible from a mobile phone which, despite its limited man-machine interaction (input and display), has better availability than the desktop PC.
5. the structure obtained can serve system-level efficiency (speed, device consumption). In our context, an implicit goal is to minimize the number of "useless" pictures that are displayed when browsing, since fast successive display of many color images through naïve timeline browsing is rather power consuming. Overall,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ACM Multimedia Conf.'2005 Singapore
Copyright 200X ACM X-XXXXX-XX-X/XX/XX ...$5.00.

the proposed structure also helps prefetch and cache images in an effective way, as we proposed in [18].

General issues of data management efficiency on mobile platforms are surveyed in [4]. More specifically, the significance of the very application addressed in this paper to the database community will be underlined in a SIGMOD'2005 keynote speech [3].

The remainder of this section outlines the principles and characteristics of the proposed contribution. We formulate the recovery of the image collection hierarchical spatio-temporal structure as an unsupervised classification task. Distinct hierarchical classifications are built for time and space, respectively from temporal and geolocation data, but are tackled with almost identical techniques. We opt for the mixture model framework, in which probabilistic models are associated both to class-conditional probability densities and to data-to-class assignments. This choice grants two advantages :

- it breaks the combinatorial explosion inherent to data grouping problems,
- it suits well the incremental nature of the task, since data-to-class assignments may evolve in a flexible way as new data streams in, using a light predict/update mechanism.

More precisely, the scheme has the following features :

1. determination of meaningful spatio-temporal groups relies on a statistical optimality criterion that exhibits several good properties with respect to the task (it finds a model complexity trade-off, it is robust to non-Gaussianity of clusters and it can cope with small samples);
2. classifications are built in an incremental manner (i.e., on-line with regarding to arrival of data) using a search procedure which adds notably split and merges to the EM algorithm, thus avoiding some poor local optima of the abovementioned criterion and enabling the update structural evolution (including the number of clusters);
3. a hierarchy of clusterings is built bottom-up and updated over time at low cost;
4. in all of these phases, we try to avoid critical arbitrary parametrization.

A necessary side-task in the real application is the determination of a small subset of "visually representative" images from the images contained in a group. We do not address this herein, as there exist effective techniques proposed in the context of video summarization (eg. [26]), that could be employed.

The remainder of this paper is organized as follows. Section 2 surveys work related to temporal and/or geolocation-based structuring for the application at hand. Section 3 then discloses the technical proposal : the probabilistic modelling and associated optimization phases on the finest-scale layer of the hierarchy are described. The process for building and tracking a hierarchy is then presented. Section 4 provides experimental results. Finally, the work is summarized and perspectives are sketched in section 5.

2. RELATED WORK

Time and geolocation annotations on personal images have been introduced in many services on the market, for the time being ignoring automated organization (Microsoft World Wide Media eXchange system [24], Picasa "Hello" (www.hello.com), Cognima (www.cognima.com)). Whether image collections should be manually or automatically organized (despite possible errors) is still under debate [17], but we advocate, with many others quoted below in this section, that automation is more beneficial than harmful, especially as it can exploit several modalities and self-evaluate its confidence level.

For existing automated schemes at research stage, time stamp has long been a favourite since it is an intuitively appealing, cheap and reliable measurement. Segmenting the sequence of time stamps has been viewed in [14, 15, 22] as the incremental detection of gaps. Some thresholding sets the definition of a "meaningful" gap. Time structuring can also be combined with image features [7, 15], or the camera settings [10]. Personal diary structuring based on location was proposed in [12], but measuring location continuously in time (rather than based on punctual picture acquisition). Partitions are extracted at multiple scales, based on a piecewise parametric trajectory model. By this means, one attempts to recover significant temporal episodes and areas. A work close in spirit is [2], although the modelling formalism differs. The recent work described in [1] proposes some elementary automated organization, but contributes mainly in the browsing mechanisms.

The work closest to the present paper is [19], which also organises an image collection hierarchically, based on time and location clusters. To our understanding, their work incorporates a series of rules derived from user expectations. Although these rules are very appropriate (especially towards joint time/space criteria), they seem to imply more arbitrary parametrization than the present scheme, where e.g. intra-cluster variability is learned. Furthermore, their scheme is not incremental, but works in batch mode. In our view, an incremental scheme appears necessary to always keep the collection organized without user needing to think about it. Running on a mobile phone as a permanent background task with low priority, the computational demand of our technique is then far less than, for instance, real-time video codecs currently running on such platforms.

3. PROPOSED TECHNIQUE

We formulate the recovery of the image collection hierarchical spatio-temporal structure as an unsupervised classification task. Distinct hierarchical classifications are built for time and space, respectively from temporal and geolocation data. The technique exposed in this paper is used for both series, almost identically. The end-user could switch between these two classifications to browse his collection, according to what better suits his/her memory or leads to faster browsing. We defer to the end of section 3.1 a further remark on this point. Alternatively, a companion paper [21] focuses on the combination of (single-scale, not hierarchical) spatial and temporal partitions into an hybrid geo-temporal representation.

3.1 Modelling and optimality criterion

We opt for the mixture model framework, in which probabilistic models are associated both to classes and data-to-class assignments. This very latter point makes it attractive

for the incremental nature of the task, since data-to-class assignments may evolve in a flexible way as new data is made available.

The data D (either location (x, y) or time t stamps) is assumed to be drawn from a random Gaussian mixture process with probability density :

$$p(D) = \sum_{k=1}^{K} p_k \mathcal{N}(D|\mu_k, \Sigma_k), \quad (1)$$

where the probabilities p_k are the mixing proportions and $\mathcal{N}(D|\mu, \Sigma)$ denotes a Gaussian distribution with mean μ and covariance Σ.

With mixture model modelling, a criterion for fair comparison between clustering hypotheses that might have different number of classes consists in comparing their integrated completed likelihoods (ICL) [5]. In contrast with the BIC criterion derived from the marginal likelihood of the data, this criterion optimizes the joint likelihood of the data D and the unobserved data-to-model assignments Z. The introduction of the latter variable accounts for the goal of discovering the hidden structure in the data. Given a clustering hypothesis H_K, the criterion is defined as :

$$p(D, Z|H_K) = \int p(D, Z|\Theta_K, H_K) p(\Theta_K|H_K) d\Theta_K \quad (2)$$

where $\Theta_K = (\theta_1, \theta_2, \ldots, \theta_K)$ is a parameter vector for H_K and $\theta_i = (p_i, \mu_i, \Sigma_i)$, $1 \leq i \leq K$. Practical computations exploit a BIC-like approximation for (2), expressed by :

$$ICL = -ML + \frac{\nu_K log(n)}{2} - \sum_{k=1}^{K} \sum_{i=1}^{n} t_{ik} \cdot log(t_{ik}) \quad (3)$$

where ML is the maximized mixture loglikelihood, ν_K is the number of independent parameters in the model with K components, n is the number of data elements and t_{ik} is the posterior probability for an observation i of originating from cluster k. These t_{ik}, supplied by the E step of the EM algorithm, are in fact expectation values of the binary assignment random variables z_{ik}.

Expression (3) is a self-explanatory variation on the BIC criterion : the additional entropic term on the right favours well-separated clusters [5]. As a practical benefit, this increases the ability of this criterion to identify correctly non-Gaussian clusters, which our goal frequently encounters. Besides, it involves only light computation, compared to alternative mixture models (eg. mixture of Student densities [6]).

Further, it is frequent that a cluster is assigned a little amount of data, leading to a poor estimate of its empirical covariance. We deal with this issue by introducing, in the M-step, regularized covariance estimates, computed as expectations of the posterior distribution of these covariance matrices (using respectively Gamma and Wishart conjugate Bayesian priors for time (one dimensional) and location (two dimensional)).

In contrast with [19], all desirable properties of the scheme are incorporated into a single probabilistic model and optimality criterion, rather than a series of rules. As a result, relevance of partitions found may be evaluated numerically, as a whole, and if no or little structure in the data (i.e. clusters) exist (or can be found), on one of the axes (e.g. location) and on some portion of time, the user can be switched automatically to the other axis (e.g. time).

3.2 Optimization of the proposed criterion

Temporal tracking of a data partition involves adjusting the data-to-cluster assignments, as well as adjusting the number of groups when new data provides evidence in this sense. Using the solution obtained at time t to initialize the local optimisation of (3) at $t + 1$ with an EM algorithm is overall a good idea : this ensures stability of the structure through which the user browsing is certainly beneficial and supplies, at no extra cost, an explicit temporal link between corresponding groups. Still, two major issues remain :

- this does not enable evolution of number of groups,
- the data stream cannot be modelled as a series of *independent* samples from a fixed probability density, in contrast with many applicative settings. As a result, the optimization hypersurface is rather shaky over time and poor local minima are often obtained if a classical conservative EM-only update is used.

A joint solution to these two issues is proposed. Briefly stated, by evaluating and (possible) applying joint split & merges among current clusters, semi-local jumps in the search space are attempted in the search space. Our approach differs significantly from the closest work [25], which does not deal with incrementality and keeps the number of groups constant. We alternate this phase (semi-local) with EM runs (local), until convergence. The process is "well-behaved", as all steps attempt to decrease the same criterion, and serves two purposes : it avoids many local minima and practically enables evolution of the number of clusters over time.

The proposed procedure is further detailed :

Split and merge criteria:

Because of the high number of split and merge possibilities, candidates operations should first be ranked. Criteria for this are proposed in [25], but they are not suitable for small samples. For example, if components under comparison that have a single observation each, they are not deemed good candidates for a merge. In our context, this configuration is frequently encountered. Alternatively, we propose the use of the following discrepancy measure, based on the Malahanobis distance, to compare components:

$$J_{merge}(i, j, \Theta) = min\{D(\mu_i, \Sigma_i, \mu_j), D(\mu_j, \Sigma_j, \mu_i)\} \quad (4)$$

where $D(\mu_j, \Sigma_j, \mu_i) = (\mu_i - \mu_j\}^T \cdot \Sigma_j^{-1} \cdot (\mu_i - \mu_j)$.

Our split criterion is based on an entropic feature of each component. Because a component with high entropy suggests that the component does not fit well its associated data, or that another model also somewhat fits this data, components are ranked for possible splitting according to the following criteria :

$$J_{split}(k, \Theta) = \sum_{i=1}^{n} t_{ik} \cdot log(t_{ik}) \quad (5)$$

Initialization of the new parameters :

Parameter initialization for the new model $\theta_{i'}$ resulting in the merge of two components parametrized by θ_i and θ_j :

$$p_{i'} = p_i + p_j \quad \text{and} \quad [\mu_{i'} \quad \Sigma_{i'}]^T = \frac{p_i[\mu_i \quad \Sigma_i]^T + p_j[\mu_j \quad \Sigma_j]^T}{p_i + p_j} \quad (6)$$

Algorithm 1 ICL Optimisation

Initialization: add the first data element and create a model M containing just one component. M^* and ICL^* are respectively the current model and its ICL criterion.
1.Add new data and run the EM algorithm with the model M until the convergence of ICL.
2.Split: rank candidates $\{S_1, \ldots, S_d\}$ for splitting according to the Mahalanobis distance.
while the FD_{max} first components are not tested **do**
- building a new model M and initialisation of its parameter Θ
- run of the EM algorithm until convergence of the ICL criterion
if $ICL < ICL^*$ **then**
$M^* \leftarrow M$ & $ICL^* \leftarrow ICL$ and go back to step 2
end if
end while
3.Merge: rank candidates $\{M_1, \ldots, M_f\}$ for merging according to their entropy. $M_k = \{i, j\}$ represents the merge of components i and j in the model M^*.
while the FD_{max} first components are not tested **do**
- building of a new model M and initialisation of its parameter Θ
- run of the EM algorithm until convergence of the ICL criterion
if $ICL < ICL^*$ **then**
$M^* \leftarrow M$ & $ICL^* \leftarrow ICL$ and go back to step 3
end if
end while

A split of component θ_k into two components $\theta_{j'}$ and $\theta_{k'}$ exploits the following initializations :

$$p_{j'} = p_{k'} = \frac{p_k}{2}, \quad \mu_{j'} = \mu_k + \epsilon, \quad \mu_{k'} = \mu_k - \epsilon \tag{7}$$

$$\Sigma_{j'} = \Sigma_{k'} = det(\Sigma_k)^{(1/d)}/I_d \tag{8}$$

where ϵ is a small vector colinear to the eigenvector associated to the largest eigenvalue of Σ_k, $det(\Sigma)$ denotes the determinant of Σ and I_d the d-dimensional identity matrix.
Overall optimization procedure
As a new data element streams in, the incremental algorithm first attempts several splits of components, followed by several merges, and finally local EM loops. If this globally improves the ICL criterion, the search move is retained. In such a case, the list of candidates for split or merge is recomputed and the procedure loops (generally, 2 to 5 times). The algorithm 1 details our algorithm. Parameter FD_{max} defines the maximum number of candidates for splitting or merging.

Overall, the proposed approach attempts a trade-off between the ability of the scheme to scan potentially good parts of the search space and computational load. Let us make the point that, as an iterative scheme, its practical cost is tightly related to amount of structural re-organization occuring within the data set, which is usual small. Besides, since this restructuring is usually local, the scheme is amenable to many classical extensions for speeding up the EM technique (e.g. partial E-steps).

3.3 Incremental hierarchical algorithm

Tree organization

This phase builds, on the incremental clustering scheme presented so far, a hierarchy of mixture models. Hierarchical mixture model-based clustering was proposed in [9], but in a batch version and for building binary trees. We extend this type of technique in several ways :

- by creating and maintaining a view on it that is a tree containing only selected levels from the binary tree, the nodes on this view having hence a variable number of children (fig. 1). More precisely, we build a binary tree, but then operate a selection among nodes, trying to avoid uninteresting and strongly redundant partitions. The ICL criterion again provides a consistent solution to the issue. Figure 1 describes this process of level selection. Proceeding from root to leaf, we search for 'local minima' in ICL, in the set of optimal partitions found at each level of the binary tree. Indeed, should there be a marked hierarchy, these local minima correspond to plausible clustering hypotheses,
- by introducing a new hierarchy update procedure, as detailed in the next paragraph.

Procedure for updating the tree

The main idea is to propagate new data from the root to a leaf, updating each level of the hierarchy and re-organizing from scratch only sub-trees where structural changes appear to be needed. As data proceeds from root to leaf and at each level, the incremental classification technique adjusts the parameters of the models and update data-to-model assignments (as described in section 3.2). The scheme, when operating at other-than-leaf level, should also cope with a practical issue : it should let the structure evolve in a flexible way (including splits and merges), but not slide towards local minima corresponding to partitions already existing at finer levels of the hierarchy. This is dealt with using the following procedure.

First, let us defined a node as 'changed' if the set of data assigned to it (in the MAP sense and not counting the last datum) has changed after EM updates following introduction of a new data element in the scheme. The process to update a node q consist in first to retrieve the model associated to the set of its children (noted c_q), and apply our ICL optimisation limiting the number of splits to one in order to not sliding to a partition existing at finer levels. This step enables to detect if the new data element involve re-structuring of the updated node. According to the modification, we apply one of the following rules:

1. if the new data is associated to an 'unchanged' com-

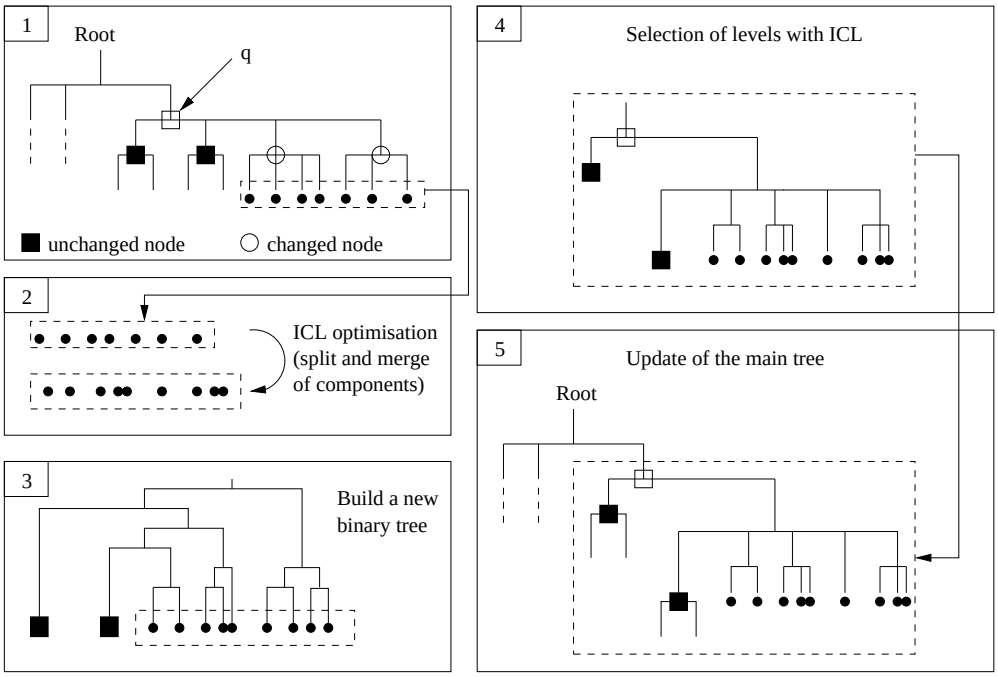


Figure 2: Update of a subtree (step 3 of our incremental hierarchical algorithm): We add the new data by the root, retrieve the model associated to its children and apply our ICL optimisation by testing splits and merges of components. In this example, no change appears and the new data element is affected to the unchanged component q. The update is then propagated to this node. Fig.1 presents the changed and unchanged nodes after the process of our ICL optimisation. We suppose here that the new data is associated to a changed component. We then retrieve the leaves of the changed nodes and applied our ICL optimisation (fig.2). We re-build a binary tree from the updated leaves and the unchanged nodes $\in c_q$ (fig.3) and select the relevant levels based on the ICL criterion. Fig.4 presents the obtained new subtree. Finally the main initial tree is updated with the subtree. Note that the children of the unchanged nodes $\in c_q$ are kept.

ponent q, the subtree under this component is simply updated, propagating EM runs downwards. If q is a leaf, we try to detail it with our ICL optimisation;

2. if the new data is associated to a new component, and that no other component is changed, this corresponds to a broadening of the current level of the hierarchy;

3. if the new data is associated to a 'changed' component, this implies more important re-structuring of the data. The different steps are then:
   - select the leaves of all the changed components
   - build a new subtree t with [9] from the selected leaves and the unchanged components $\in c_q$
   - optimize the tree with our levels selection
   - update the main tree with the new subtree
   Figure 2 shows an example of the step 3.

An interesting property of our technique is that the parameters of the tree are automatically determined by our algorithm: the son's number, the width and the depth are free. The selection of pertinent levels enables to speed up our algorithm by limiting the depth of the obtained tree and improving its robustness face to modification due to the add of new data. Indeed, each level being composed of distinct classes, this selection of level enables to increase the "independence" of each component and helps to decrease the number of modified components for each new data element. Moreover, this selection garanties the relevance of each levels in accordance with the browsing task.

4. EXPERIMENTAL RESULTS

We propose here to build spatial and temporal hierarchical classifications of a personal image collection composed of 721 pictures taken along 3 years. The user took pictures in France essentially, in USA and Canada. Time meta-data were directly included by the camera in each image (exif meta-data) and the location was added manually based on the real location.
Figures 3 and 4 present respectively the temporal and spatial metadata of the collection. The spatial metadata present specific characteristics since users take pictures in specific location: it is essentially composed of locations concentrated in one point.

We first examine the temporal structure obtained. Its topology is provided in Figure 6 and Figure 5 shows several obtained trees all along the process of temporal classification. Finally selected zooms displaying the structure superimposed on the original data are depicted in Figure 7.

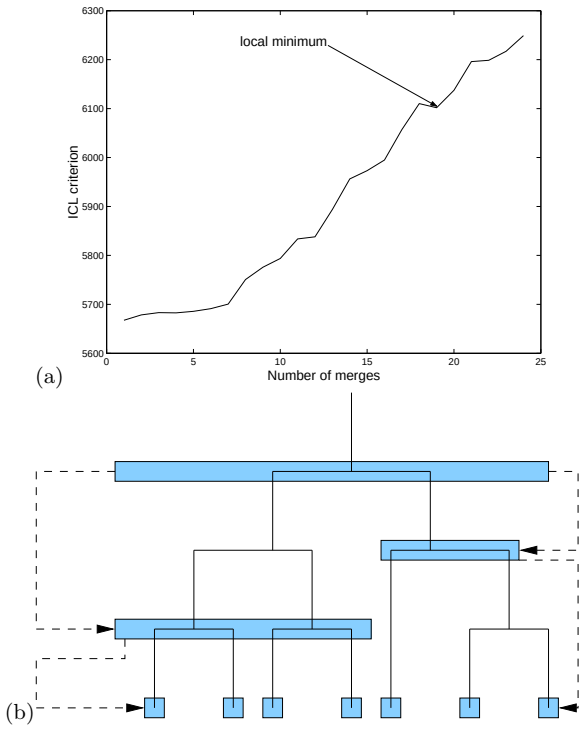


Figure 1: Selection of levels corresponding to local optima of the ICL criterion: (a) the optimal ICL criterion found at each level of the binary tree represented on (b) is plotted. The grey rectangles indicate the corresponding selection of partitions. Once an optimum is found at a level q, we search for another local optima in each subtree from q. 'Local' minima here is to be interpreted as follows : both slightly coarser and slightly finer partitions are worse, in the ICL sense.

The tree obtained is composed of 4 levels and is well-balanced. The number of children per node varies from 2 to 14. We noticed on Figure 5 that our classification extends in depth and width as new data are added. The trees (b), (c), (d) and Figure 6 present similarities (dashed squares represent the similar sections). The stability of the obtained trees all along the classification process seems continuous in time. A new image generally does not involve a lot of modifications.

Figure 7 presents partitions obtained at various levels of our tree. Figure 7(a) shows the coarsest level. All components are well delimited, providing relevant summaries of the collection. Components 2,3 and 4 on Figure 7(a) are respectively detailed in Figure 7(d), (b) and (c). Children of the components 4 and 3 provide well-defined partitions since all the temporal gap are correctly emphasized. For component 2, meaningful temporal episode are found but we notice over-segmentation, as groups 2.9 and 2.10, certainly due to a larger evidence of small sample associated to one component.

First experiments consist in classifying directly all the spatial metadata and provide poor results due to their characteristics. The user often took pictures in a same place, involving compact clusters with many data concentrated in one point. We obtained just one level with compact clusters for each location. Our algorithm then failed to provide a good summarization of the collection. Due to the data structure, our hierarchical algorithm failed to find coarser levels.

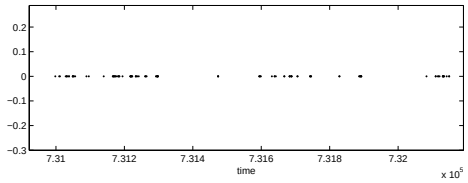


Figure 3: Real scenario: temporal metadata of the image collection. The dots represent the temporal metadata.

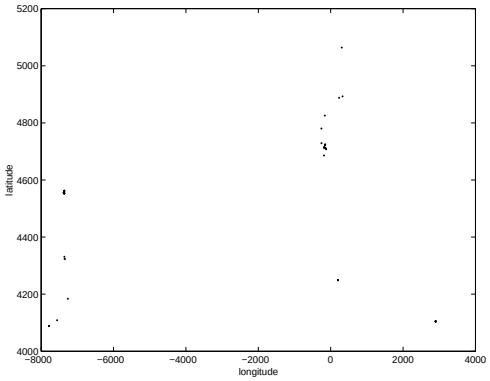


Figure 4: Real scenario: spatial metadata of the image collection. The dots represent the spatial metadata. It is composed essentially of locations concentrated in one point.

To prevent from this kind of configuration, we summarize the spatial metadata by keeping just one value of each distinct coordinate (x, y). So that one coordinate (x, y) can represent the location of several images. This summarization process provides 135 distinct locations to classify.

Figure 8 and 9 show respectively the hierarchical spatial classification and examples of obtained partitions at different levels.

The tree obtained is composed of 3 levels and the number of children per node is moderate, varying from 2 to 6. The coarser level is presented in Figure 9(a), and Figure 9(b) provides a zoom on components 1, 2 and 3. Components are well-distinct and compact due to the characteristic of the data. Our optimization algorithm have a tendency to regroup isolated data together, as shown with component 3. This aspect can nevertheless be relevant for a browsing perspective. Children of component 2 and 7 are respectively presented in fig.9(d) and (c). Both obtained partitions are quite relevant since the different main locations are found.

To evaluate practically the obtained hierarchical classification, we use the same heuristic as in [20] which are the

precision and recall for the detected event boundaries:

$$precision = \frac{\text{correctly detected boundaries}}{\text{total number of detected boundaries}} \quad (9)$$

$$recall = \frac{\text{correctly detected boundaries}}{\text{total number of ground truth boundaries}} \quad (10)$$

The user manually finds 58 events in its collection. Note that he generally regroups holiday pictures or successive occasional events in a same component (both taken on several weeks). To compare with our result, we retrieve all the leaves of our temporal classification to obtain our finer classification. It is composed of 107 components and we obtained 57 correct boundaries: $recall = 98\%$. We succeed to get back all the events in the collection with a very good accuracy. Since we provide a hierarchical classification, we do not compute the precision with the finer partition. We propose to check in the temporal tree if all the finer nodes are correctly regrouped in an appropriate subtree (if all the components of the manual partition are represented by a node in our tree). We found 50 manual components correctly regrouped in distinct nodes: $precision = 86\%$. The errors is related to holidays or occasional successive events which are divided in separate leaves.

According to the user, 25 distinct locations are present in the collection. We found successfully 23 locations with their associated images, so the spatial partition succeed to emphasize the main location of the collection. Two errors remain: one leaf regroups two close locations and one location is divided into two distinct nodes. This last case is due to images taken during a stroll where the spatial data are badly structured. The obtained hierarchy is also satisfactory since all the components regroup related locations. For example, the component 2 represents all the different locations in the home city of the user while component 1 is associated to surrounding areas.

The trade-off between temporal structural flexibility and computational load can be evaluated as follows. During the process of our algorithm on the temporal data, the agglomerative algorithm regenerating a binary tree has been called 25% of the time and, for each call, concerned on average 27% of the data. For the spatial data, it was called 60% of the time and concerned on average 8% of the data (in the first iteration, the agglomerative algorithm is often called due to the lack of data stability).

5. SUMMARY AND PERSPECTIVES

This paper proposes a technique to organize a personal image collection acquired from a mobile imaging device, geographically and temporally, since this is both useful and a low-cost, technically feasible way of recovering the 'events' that are meaningful to users. The main requirements in this study were to design a fully automatic technique avoiding troublesome arbitrary parametrization and to rely solely on the data (i.e. not use geographical information systems). The overall idea is that a hierarchy of mixture models is being tracked, as new data enters the system. The integrated completed likelihood criterion was used to maintain an uniform definition of partition quality and a clear separation is done between modelling and optimization. The probabilistic nature of assignments can handle flexible re-allocation of data to clusters, and coupling local to semi-local avoid most poor local minima. By nature, mixture model and EM scale up well to large amounts of data. The scheme is also shown to be (to some extent) robust to non-Gaussianity of clusters and small samples. Let us point out that the computation of the incremental algorithm is by nature distributed over time (several days) as a background task, thus consumming few resources compared to other activities of mobile devices. While the focus of the paper is kept on the structuring phase, the output is dedicated to a browsing navigation interface on a mobile device. We are currently examining how to make better joint use of temporal and spatial data, given the confidence of local sections of the partitions obtained. The ICL measures may be used to this end, but the task is particularly challenging when addressing multiple scales of the hierarchy that do not necessarily correspond in time and space.

In the more general landscape of current issues in multimedia document indexing, we believe the needs addressed in this paper are important practical stakes on mobile devices, and the solution quite representative of needed interaction between pattern recognition issues and system-level data management.

6. REFERENCES

[1] A. Aris, J. Gemmel, and R. Lueder. Exploiting location and time for photo search and storytelling in MyLifeBits. Technical Report MSR-TR-2004-102, Microsoft research, Sept. 2004.

[2] D. Ashbrook and T. Starner. Learning significant locations and predicting user movement with GPS. In *IEEE Int. Symp. on Wearable Computing (ISWC'2002), Seattle, USA*, pages 101–108, Oct. 2002.

[3] G. Bell. Mylifebits : a memem-inspired personal store : another tp database. In *Proc. of SIGMOD'2005*, Baltimore, USA, June 2005.

[4] G. Bernard, J. Ben-Othman, L. Bouganim, G. Canals, B. Defude, J. Ferrié, S.Gançarski, R. Guerraoui, P. Molli, P. Pucheral, C. Roncancio, P. Serrano-Alvarado, and P. Valduriez. Mobilité et bases de données : état de l'art et perspectives. *TSI*, (3/4), 2003.

[5] C. Biernacki, G. Celeux, and G. Govaert. Assessing a mixture model for clustering with the integrated classification likelihood. In *IEEE Transaction on pattern analysis and machine intelligence*, volume 22, pages 719–725, Jul. 2000.

[6] C. Bishop and M. Svensen. Robust Bayesian mixture modelling. In *Proceedings Twelfth European Symposium on Artificial Neural Networks*, pages 69–74, Bruges, Belgium, Apr. 2004.

[7] M. Cooper, J. Foote, A. Girgensohn, and L. Wilcox. Temporal event clustering for digital photo collections. In *Proc. ACM Multimedia*, pages 364–373, Nov. 2003.

[8] M. Davis and R. Sarvas. Mobile media metadata for mobile imaging. In *IEEE International Conference on Multimedia and Expo (ICME 2004) Special Session on Mobile Imaging*, Jun. 2004.

[9] C. Fraley. Algorithms for model-based Gaussian hierarchical clustering. *SIAM Journal on Scientific Computing*, 20(1):270–281, 1999.

[10] U. Gargi, Y. Deng, and D. R. Tretter. Managing and searching personal photo collections. Technical Report HPL-2002-67, HP Laboratories, Palo Alto, Mar. 2002.

[11] M. Gelgon. Using face detection for browsing personal slow video in a small terminal and worn camera context. In *IEEE. Int. Conf. on Image Processing (ICIP'2001)*, pages 1062–1065, Thessaloniki, Greece, Sept. 2001. IEEE Signal Processing society.

[12] M. Gelgon and K. Tilhou. Structuring the personal multimedia collection of a mobile device user based on

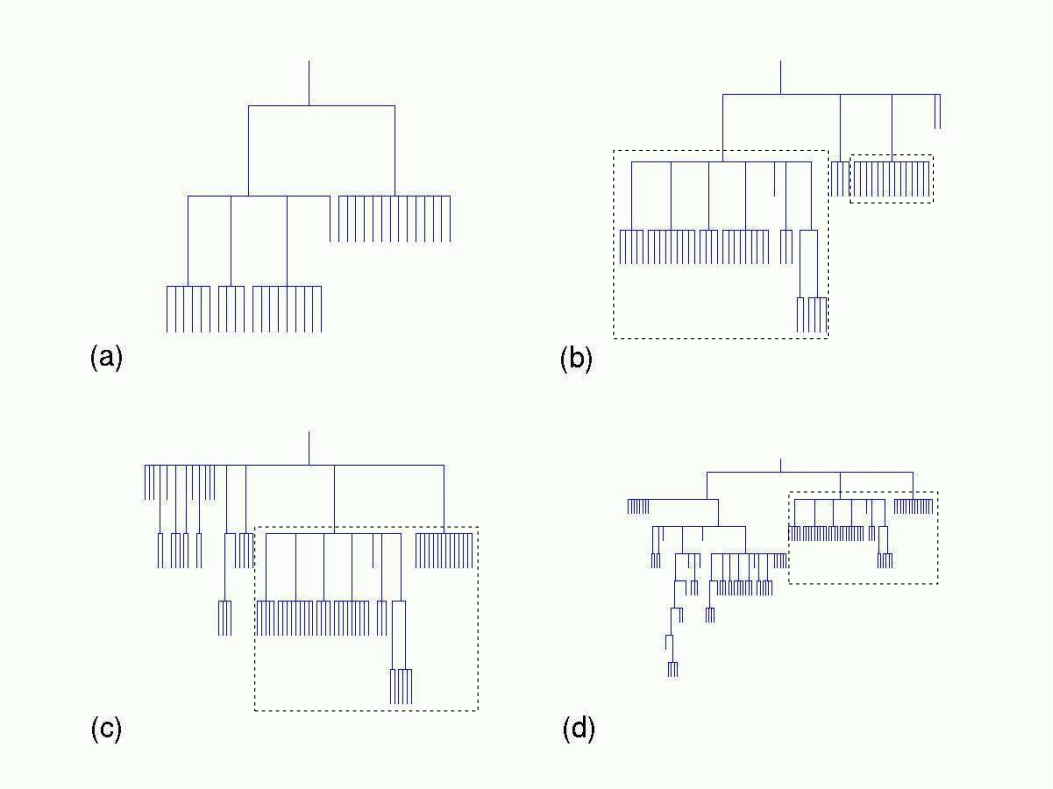


Figure 5: Real scenario: temporal hierarchical classification obtained each 150 data. We noticed that our classification extend in depth and width as data are added. Dashed squares indicate the similar section of the trees all along the classification process.

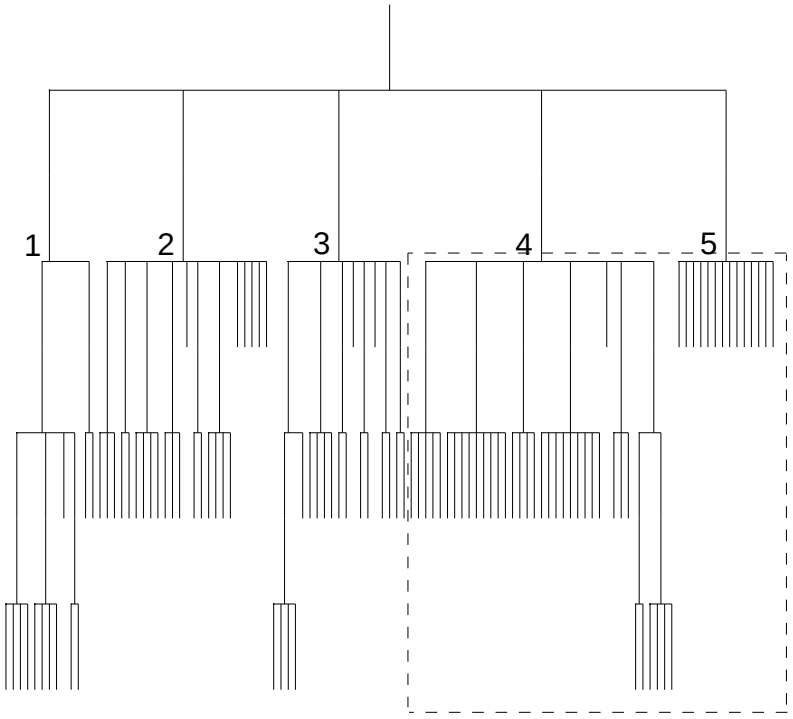


Figure 6: Real scenario: temporal hierarchical classification. Each node is named with a number. Number are arbitrary as an indication to correspondance to fig.7. Dash line represents the similar section with the obtained tree in fig.5(d). The tree obtained seems rather well-balanced since it presents a good depth-width ratio. And the number of children per node is moderate, typically from 2 to 14. This properties are encouraging for a browsing perspective.

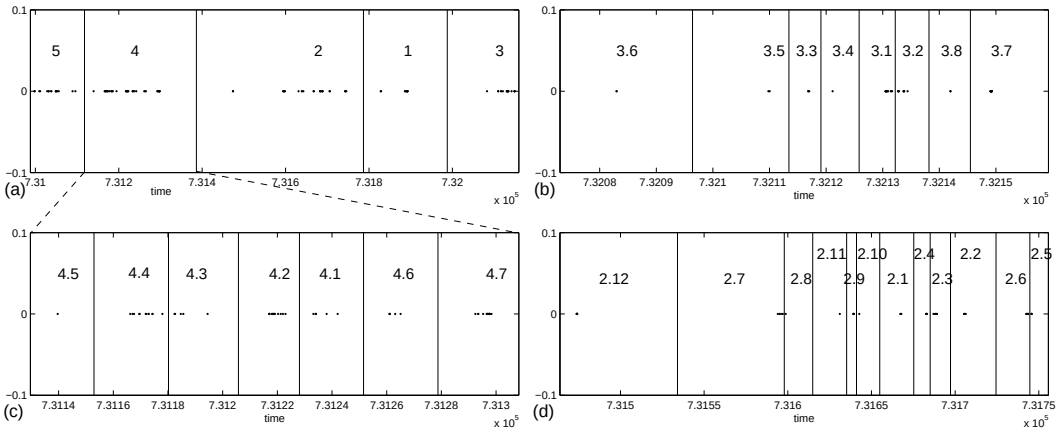


Figure 7: Real scenario: example of partitions obtained in the temporal hierarchical classification. Each number represents the components of the associate node in figure 6. Solid lines represent the boundary of components and the dots are the temporal metadata. Figure (a) represents the coarser level of our obtained temporal tree and fig.(b), (c) and (d) show respectively the children of components 3, 4 and 2. Obtained partitions present distinct clusters with visually justified boundaries. We notice over-segmentation (for example component 2.9 and 2.10).

geolocation. In *IEEE Int. conf. on Multimedia and Expo (ICME'2002)*, pages 248–252, Lausanne, Switzerland, Aug. 2002.

[13] J. Gemmell, R. Lueder, and G. Bell. The mylifebits lifetime store. In *Proceedings of the 2003 ACM SIGMM workshop on Experiential telepresence*, pages 80–83, Nov. 2003.

[14] A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *ACM Joint Conference on Digital Libraries JCDL*, pages 326–335, Jun. 2002.

[15] A. Loui and A. E. Savakis. Automatic image event segmentation and quality screening for albuming applications. In *IEEE Proceedings Int. Conf. on Multimedia and Expo (ICME'2000)*, pages 1125–1128, New York, USA, Aug. 2000.

[16] J. Luo, A. Savakis, and A. Singhal. A Bayesian network-based framework for semantic image understanding. *Pattern Recognition*, 38(6):919–934, June 2005.

[17] A. Myka. Nokia lifeblog - towards a truly personal multimedia information system. In *Proc. of Workshop des GI-Arbeitkreises "Mobile Datenbanken and Informationsysteme"*, Karlsruhe, Germany, Feb. 2005.

[18] A. Myka, J. Yrjänäinen, and M. Gelgon. Enhanced storing of personal content. US Patent 16660/10502275, Nokia corp., Jul. 2004.

[19] M. Naaman, Y. J. Song, A. Paepcke, and H. Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. In *Proc.of ACM/IEEE Conference on Digital libraries (JCDL'2004)*, pages 53–62, Jun. 2004.

[20] M. Naaman, Y. J. Song, A. Paepcke, and H. Garcia-Molina. Automatically generating metadata for digital photographs with geographic coordinates. In *International World Wide Web Conference archive, Alternate track papers & posters of the 13th international conference on World Wide Web*, pages 244–245, May 2004.

[21] A. Pigeau and M. Gelgon. Incremental statistical geo-temporal structuring of a personal camera phone image collection. In *Proc. of Int. Conf. on Pattern Recognition*, volume 3, pages 878–881, Cambridge, U.K., Aug. 2004.

[22] J. C. Platt and B. A. F. M. Czerwinski. PhotoTOC: Automatic clustering for browsing personal photographs. Technical Report MSR-TR-2002-17, Microsoft Research, Feb. 2002.

[23] K. Rodden. How do people manage their digital photographs? In *ACM Conference on Human Factors in Computing Systems*, pages 409 – 416, Fort Lauderdale, Apr. 2003.

[24] K. Toyama, R. Logan, A. Roseway, and P. Anandan. Geographic location tags on digital images. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 156–166, Berkeley, CA, USA, Nov. 2003.

[25] N. Ueda, R. Nakano, Z. Gharhamani, and G. Hinton. SMEM algorithm for mixture models. *Neural computation*, 12(9):2109–2128, 2000.

[26] P. Vermaak, J. Perez and M. Gangnet. Rapid summarization aned browsing of video sequences. In *Proc. of British Machine Vision Conference (BVMC'2002)*, pages 145–151, Cardiff, U.K., Sept. 2002.

[27] P. Viola and M. Jones. Robust real-time object detection. In *Proc. of Int. Workshop on Statistical and Computational Theories of Vision - Modeling, Learning, Computing, Samplinge (with ICCV'2001)*, Vancouver, Jul. 2001.

[28] W. Wagenaar. My memory : a study of autobiographical memory over six years. *Cognitive psychology*, 18:225–252, 1986.

[29] A. Wilhelm, Y. Takhteyev, R. Sarvas, N. Van House, and M. Davis. Photo annotation on a camera phone. In *Proc. of ACM Computer Human Interaction (CHI'2004)*, pages 234–238, Vienna, Austria, 2004.

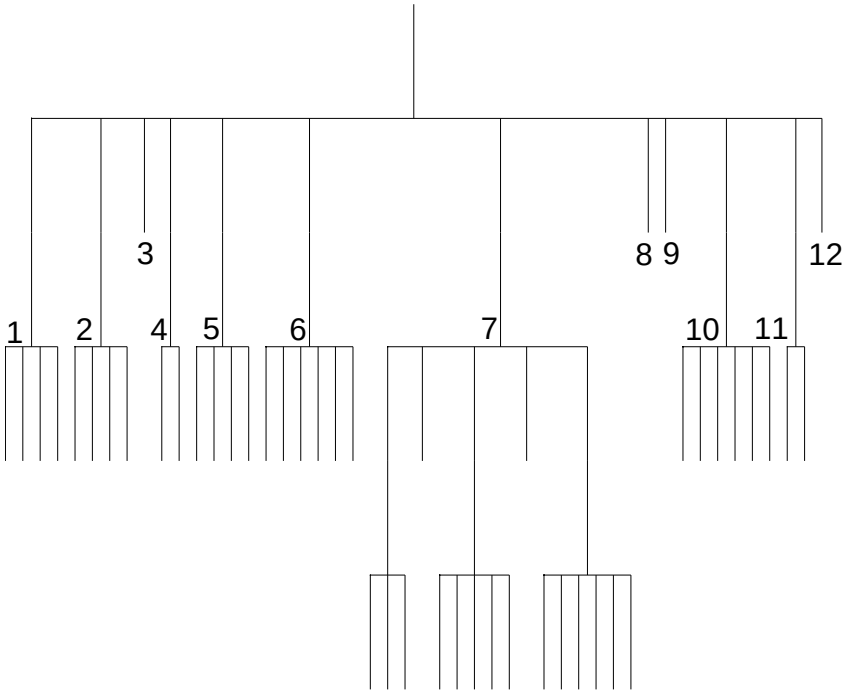


Figure 8: Real scenario: obtained spatial hierarchical classification. The obtained tree is well balanced and the number of children per nodes varies from 2 to 6. The coarser level is presented in figure fig.9(a). Numbers indicate a correspondence of branches to fig.9.

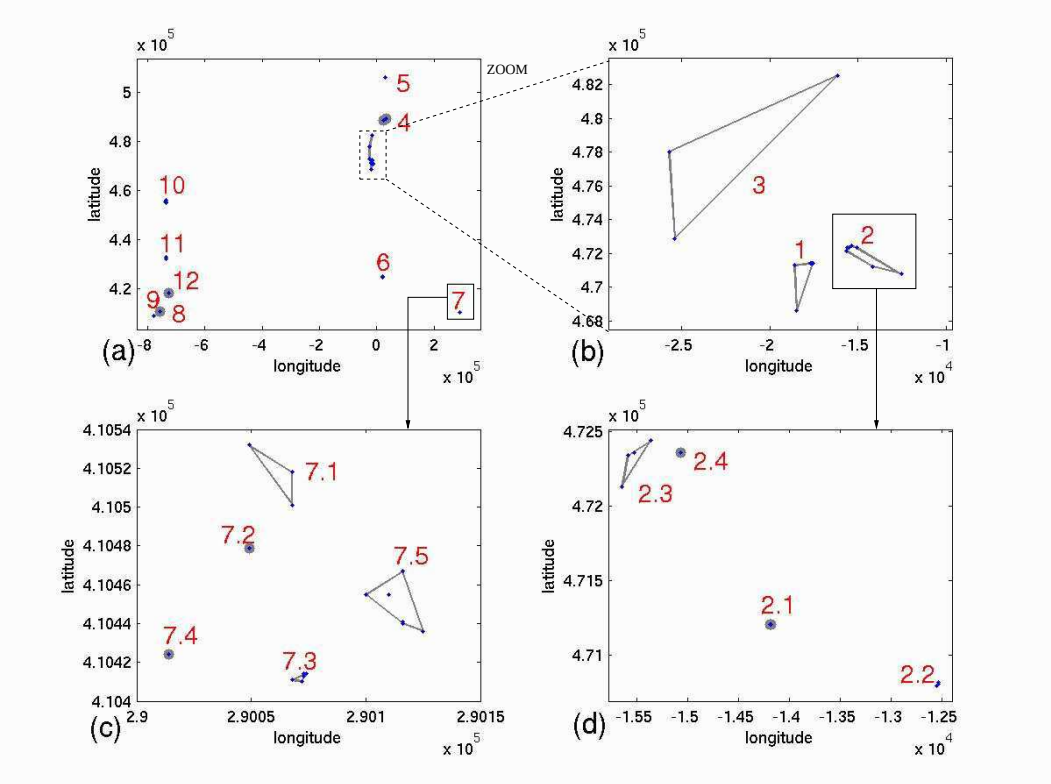


Figure 9: Real scenario: example of partitions obtained in the spatial hierarchical classification. Each number represents the components of the associate node in figure 8. The dots are the spatial metadata and thick lines are the convex hull of each component based on the maximum probability a posteriori. The arrows represent the parental relation between the partitions. Figure (a) represents the coarser level of the classification and fig.(b) is a zoom on this partition. Fig.(c) and (d) are respectively the children of the components 7 and 2.

Published in *Image and Vision Computing*, Elsevier, jan 2005.

Recovery of the trajectories of multiple moving objects in an image sequence with a PMHT approach

Marc Gelgon[a], Patrick Bouthemy[b] and Jean-Pierre Le Cadre[c]

[a]LINA / Ecole Polytechnique de l'Université de Nantes
rue C.Pauc 44306 Nantes cedex, France

[b]IRISA/INRIA [c]IRISA/CNRS
Campus universitaire de Beaulieu
35042 Rennes cedex, France

e-mail : Marc.Gelgon@polytech.univ-nantes.fr, bouthemy@irisa.fr, lecadre@irisa.fr
Tel : (33) 2.40.68.32.57 Fax : (33) 2.40.68.32.32

Abstract

This paper is concerned with the tracking of multiple moving objects in an image sequence and the reconstruction of the entire trajectories of these objects all over the sequence. More specifically, we address the joint issue of trajectory estimation and measurement-to-trajectory associations, which is the key problem in that context due to the occurrence of object occlusions or crossings. An original and efficient scheme is proposed, that adapts the Probabilistic Multiple Hypothesis Tracking (PMHT) technique to the case of tracking of regions in video, for which geometry and motion models can be introduced. Moreover, reliable partial associations can be obtained as an initialization. Data association and trajectory estimation are conducted within a probabilistic framework. The latter relies on Kalman filtering, while the former is solved with an EM algorithm for which a suitable initial configuration can be defined. The proposed tracking method is validated by experiments carried out on real image sequences depicting complex situations.

Keywords

Multiple object tracking, trajectory reconstruction, data association, EM algorithm, PMHT.

1 Problem statement

This paper is concerned with the tracking of multiple moving objects in an image sequence and the reconstruction of the entire trajectories of these objects all over the sequence. More specifically, we address the joint issue of trajectory estimation and measurement-to-trajectory associations. This is the key problem in that context due to the occurrence of object occlusions or crossings.

In video content analysis, whether for interpretation, indexing or coding, trajectories of objects - manipulated as regions in images - are of much importance. For instance for surveillance purposes, trajectories of mobile objects are generally of key interest. It may occur, however, events (temporary misdetection, occlusions, crossings) from which important ambiguities in the association of successive measurements to a track can arise.

We specify the addressed problem by describing hereunder the input data to the algorithm designed in this paper. We are provided with a batch of motion segmentation maps using an approach presented in [20], of which Fig. 1 shows an example. This technique supplies a motion-based partition of images, in which the motion region homogeneity criterion is expressed by a 2D parametric motion model. Motion estimation is supplied by a multiresolution, robust estimator and the segmentation problem is expressed and solved as the statistical estimation of a pixel label map, within a Markov Random Field framework. The set of measurements (at each time instant), includes:

- the 2D spatial supports of the extracted moving regions ;
- the estimates of motion of these regions, i.e. the 2D parametric motion models estimated between the current frame and the next one associated to these regions;
- the regions labels, i.e., their numbers (symbolic information).

The motion segmentation algorithm employed has the property that if the same region (object) is continuously extracted in successive frames, the region label is maintained. This provides a short-term temporal link which we will assume reliable (e.g., as shown in Fig. 1, identity of the two labels is relevant over images b_0 to b_6). However, since an object may temporarily be static or totally occluded, there may be lacks of detections that break that temporal link. This introduces the concept of partial trajectory. When the region reappears and is segmented again, it then bears a new label, provided by the motion segmentation algorithm (as illustrated in Fig. 1 from images b_{24}). Our focus is on determining and associating partial trajectories of regions and jointly estimating the complete trajectories of these regions, while dealing with occlusion or crossing situations.

Besides, the silhouette of the extracted region is often affected by perturbations compared to the true projection of the object in the image. Moving shadows may enlarge the expected support, while partial occlusion may cause some pixels to miss. For instance, in the sequence displayed in Fig. 1a, the total occlusion (images 14 to 23) is preceded and followed by partial occlusions of the two moving elements. As illustrated in Fig. 1*b*, this has an obvious effect on the supplied motion segmentation maps.

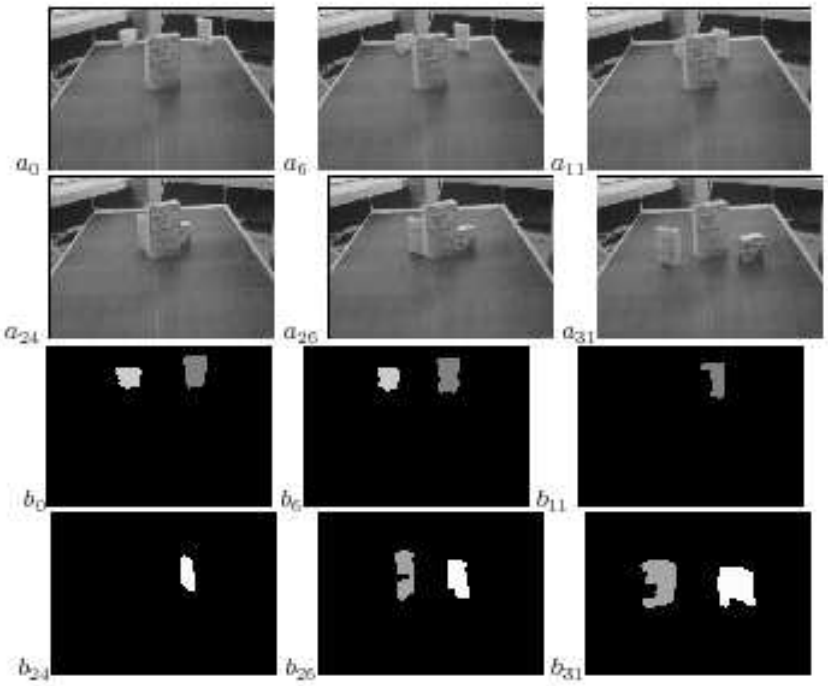


Figure 1: Original images (a) and resulting motion segmentation maps (b) at time t=0, t=6, t=11, t=24, t=26 and t=31. In this lab sequence, two moving boxes cross behind a third (static) one.

The desired output of the algorithm is two-fold :

- the correct association of the segmented regions over time, i.e., grouping of partial tracks;
- the complete trajectory of all the moving objects over the entire processed sequence, i.e. an estimated position of the object projections at each time instant (including at those when no measurement was initially available).

A core difficulty is that these two problems are tightly intricate. We briefly review below existing approaches for tracking, focusing on the issue of temporal data association.

2 State-of-the-art

Important research efforts in computer vision have been devoted to tracking objects in image sequence. In the case of region tracking, techniques based on active contours [2] or level-sets [21] have been employed, difficulties related to initialization and changes in topology being better handled by the latter approaches. It is insightful to distinguish between techniques that use a prediction and adjustment mechanism to track the image primitives, hence establishing a natural link between successive measurements and estimating model-based trajectories [14, 16, 20, 29], from those that determine merely correspondence between primitives, and thus need to address an explicit data association problem (e.g., [18]).

Data association refers to the task of identifying, for each measurement, from which physical source (moving object, in our computer vision context) it arises. Potential association

ambiguities and difficulties naturally appear when a scene contains several such physical elements. A similar issue is also encountered in general unsupervised classification tasks, but *data association* is the coined term when facing specific issues pertaining sequential data processing.

Explicit handling of the data association problem has received much attention, for a long time in the context of radar and sonar [7], more recently in computer vision. In the latter field, it has been applied to corners [4], segments [32], and regions [16, 22]. Trajectory estimation and data association problems are known to be two tightly interwoven problems. Indeed, the association between observations and objects depends on the estimated trajectories, which in turn should be computed from the whole set of measurements corresponding to a single physical element. The point is that this intricate issue is an NP combinatorial one.

A survey of data association techniques may be found in [3]. The measurement-to-trajectory model assignment can be hard, as in Multiple Hypothesis Tracking (MHT) algorithms [1, 24, 25]. Overall, MHT techniques consist in enumerating possible assignments and evaluating the pertinence of the trajectories formed, while introducing criteria to prune the assignment hypothesis tree, which otherwise would exponentially grow. Another classical tool for trajectory estimation/data association is the Joint Probabilistic Data Association Filter (JPDAF) [1], used for instance in [22] for region tracking. It is rooted in the Probabilistic Data Association Filter (PDAF) which, in e.g. Kalman filtering, updates the states using a combination of several competing measurements. The JPDAF is an enhanced version which, when there exists several such tracking processes, enforces some mutual exclusion in associations to prevent several trackers from fitting the same data. However, the JPDAF is rather a track updating technique.

In this paper, we propose an original approach relying on the Probabilistic Multiple Hypothesis Tracking technique (PMHT), which offers an attractive alternative to these classical techniques. Initially proposed in [28], a collection of works pertaining to the PMHT technique, and presenting variations thereof, may be found in [27]. They have been primarily explored in the radar and sonar domains. The statistical PMHT method consists in performing a MAP (Maximum A Posteriori) estimation of the models using Kalman filtering in the case of linear measurements and the EM algorithm for assigning, in a probabilistic manner, measurements to trajectory models. A key point is that doing so, it avoids the NP-hard combinatorial issue, in particular inherent in MHT techniques. We refer the reader to [8, 27, 28] for in-depth coverage.

In [17], the authors propose a recursive scheme closely related to PMHT in which the association variables form a Markov random field. The method we have designed remains, as in [28], with a batch approach, and a preliminary version was described in [9]. In [10], a modification was introduced to the PMHT, with a similar viewpoint to ours, so as to exploit the prior knowledge given by the existence of partial tracks, by constraining certain sets of measurements to be assigned to a single track.

A major aspect of target tracking with trajectory reconstruction is the modelling, of the state temporal evolution and of the relation between state and measurements. In many naval surveillance scenarios, piecewise linear trajectories are assumed, while airborne applications usually require more flexible manoeuvering models. A classical solution is to employ Kalman filtering with dynamic and measurement models that are fixed in their form and parametrization [16]. We shall also take this approach. Recently, Hue et al. [12] have proposed a promising improvement on PMHT on this latter aspect, by introducing particle filtering (also known as Condensation or bootstrap filter [13]) which, compared to the abovementioned model, makes weaker assumptions on the form of the dynamic and observation processes. Flexibility in the dynamic process modelling has also recently been introduced in [31].

Applications of PMHT can so far be found in radar and sonar [8] and high-energy particle physics [26]. Still, to our knowledge, point-wise measurements are generally considered. Im-

portant contributions of the present work consist, besides demonstrating the effectiveness of PMHT for a common computer vision problem, in proposing the following adaptations :

- spatial extent (2D region support) and velocity information are properly incorporated into the PMHT scheme,
- a dedicated and efficient initialization is provided.

The remainder of the paper first presents the manner in which we model the problem, fitting in the PMHT framework (Section 3). We then recall how this category of problems may be solved using the Expectation-Maximization algorithm (Section 4). Section 5 presents the extension of the PMHT approach we have designed to handle tracking in video (in particular, initialization of the EM algorithm). Section 6 provides experimental results, and in section 7 we draw some concluding remarks.

3 Modelling of the problem

A *measurement* in our problem is a set of elements describing a segmented region at a given image instant, as listed in Section 1. They will be more formally defined hereafter. We shall call *partial track* a set of successive measurements linked over time by identity of the label attached to their corresponding regions. The goal is to recover *entire tracks* over the whole image sequence, each entire track being issued from the set of measurements corresponding to the same single physical moving object. To each partial track is associated a *2D trajectory model* of the mobile element, to be estimated from the measurements.

Let us denote $\mathcal{Z}$ the set of observed measurements $Z(t)$ in the batch $[t = 0, \ldots, t = T]$ corresponding to the processed image sequence. At each time instant t, $Z(t)$ is composed of a set of s_t measurements $z_j(t)$. They will be instanciated hereafter. We have :

$$\mathcal{Z} = \Big[Z(1), \ldots, Z(T)\Big] \tag{1}$$

$$Z(t) = \Big\{z_1(t), \ldots, z_{s_t}(t)\Big\} \tag{2}$$

We assume that measurements originate from $\mathcal{M}$ moving objects in the scene. As $\mathcal{M}$ is unknown (and to be determined), the algorithm works throughout considering M trajectory models, where M is the number of partial tracks ($M > \mathcal{M}$). In a second stage, $\mathcal{M}$ will be determined by identifying redundant trajectory models among the M ones.

Each of the M trajectory models is described by a time-dependent state vector, and an evolution model of this state vector. Let us denote $x_m(t)$ the state vector of trajectory model m at time t. We also define the set $X(t)$ of state vectors at a given time t and their set $\mathcal{X}$ over the batch as follows :

$$\mathcal{X} = \Big[X(1), \ldots, X(T)\Big] \tag{3}$$

$$X(t) = \Big\{x_1(t), \ldots, x_M(t)\Big\} \tag{4}$$

Each region is represented by two elements :

- a geometric (polygonal) model of its contour. The polygonal approximation employs the technique described in [30];

- its kinematics, described by a 2D affine inter-frame motion model. Let us recall that a 2D affine motion model is defined as follows :

$$\omega_\theta(p) = [a_1 + a_2 x + a_3 y, a_4 + a_5 x + a_6 y]^T \tag{5}$$

where $p(x, y)$ is an image point, $\theta = [a_1, a_2, a_3, a_4, a_5, a_6]^T$ and $\omega_\theta(p)$ is the velocity vector given by the considered motion model at point p.

The state vector $x_m(t)$ and the measurement vector $z_j(t)$ are hence made up of two components:

$$x_m(t) = \left[\begin{array}{c} \mathcal{G}_m(t) \end{array} , \begin{array}{c} \Theta_m(t) \end{array} \right]^T \qquad m = 1, \ldots, M \tag{6}$$

$$z_j(t) = \left[\begin{array}{c} \tilde{\mathcal{G}}_j(t) \end{array} , \begin{array}{c} \tilde{\Theta}_j(t) \end{array} \right]^T \qquad j = 1, \ldots, s_t \tag{7}$$

where

- $\mathcal{G}_m(t) = \{P_1^m(t), \ldots, P_{n(t)}^m(t)\}$ and $\Theta_m(t) = [a_1^m(t) \ , \ \ldots \ , \ a_6^m(t)]^T$ are respectively the geometric (i.e., the $n(t)$ vertices of the polygonal shape representing the region) and kinematic component of the state vector (i.e. the six parameters of the affine motion model);
- $\tilde{\mathcal{G}}_j(t) = \{\tilde{P}_j^1(t), \ldots, \tilde{P}_j^{\tilde{n}(t)}(t)\}$ is an ordered set of $\tilde{n}(t)$ vertices resulting from the polygonal approximation of the segmented region at time instant t;
- $\tilde{\Theta}_j(t) = [\tilde{a}_j^1(t) \ , \ \ldots \ , \ \tilde{a}_j^6(t)]^T$ is the estimated parameter vector of the affine motion model, obtained with the multiresolution robust estimation method described in [19].

We assume that the temporal evolution of each component of the state vector $x_m(t)$ can be appropriately represented by a first order model, with additive Gaussian white noise. Besides, we consider that the measurements are corrupted by an additive Gaussian white noise, which covariance matrix is denoted R_m.

Kinematic component

The parameters of the motion model Θ_m are considered decorrelated and are estimated independently. A classical first order evolution model is selected for these parameters. It is expressed by relation (8) for any r^{th} parameter ($r = 1, \ldots, 6$) :

$$\begin{bmatrix} a_r^m(t+1) \\ \dot{a}_r^m(t+1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_r^m(t) \\ \dot{a}_r^m(t) \end{bmatrix} + \begin{bmatrix} \epsilon_{1,r}^m(t) \\ \epsilon_{2,r}^m(t) \end{bmatrix} \tag{8}$$

where $[\epsilon_{1,r}^m, \epsilon_{2,r}^m]^T$ is a Gaussian random vector, which covariance matrix Q_ϵ is expressed as :

$$Q_\epsilon = \sigma_\epsilon^2 \begin{bmatrix} \frac{1}{3} & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix} \tag{9}$$

The measurement equation is defined by stating that an additive Gaussian measurement noise $\eta_m^r(t)$ of variance σ_η^2 affects each motion parameter :

$$\tilde{a}_r^m(t) = a_r^m(t) + \eta_r^m(t) \qquad (r = 1, \ldots, 6) \tag{10}$$

Considering we have no prior knowledge on the kinematics of the moving object, no training set, and that no reliable estimation of the measurement uncertainty is available, σ_ϵ^2 and σ_η^2 are empirically user-set parameters.

Geometric component

The geometric model is formed by the set of vertices of the polygon approximating the region boundary. The temporal evolution of each of these vertices is designed by involving the affine motion model $\widehat{\Theta}_m(t)$ estimated on the region m and filtered over time. We have, for any vertex :

$$P_q^m(t), q = 1, \ldots, n(t) \ : \ P_q^m(t+1) = P_q^m(t) + \omega_{\dot{\theta}m(t)}(P_q^m(t)) \tag{11}$$

If we denote $P^m(t) = [u_q^m(t), v_q^m(t)]^T$ the temporal evolution model for the geometric component is specified by :

$$\begin{bmatrix} u_q^m(t+1) \\ v_q^m(t+1) \end{bmatrix} = \begin{bmatrix} a_0^m(t) \\ a_1^m(t) \end{bmatrix} + \begin{bmatrix} 1 + a_2^m(t) & a_3^m(t) \\ a_4^m(t) & 1 + a_5^m(t) \end{bmatrix} \begin{bmatrix} u_q^m(t) \\ v_q^m(t) \end{bmatrix} + \begin{bmatrix} \zeta_{q,1}^m(t) \\ \zeta_{q,2}^m(t) \end{bmatrix} \tag{12}$$

where the $\zeta_{q,1}^m(t)$ and $\zeta_{q,2}^m(t)$ are drawn from Gaussian distributions, which covariance matrix Q_ζ is expressed as :

$$Q_\zeta \ = \ \sigma_\zeta^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{13}$$

The relation between the geometric model and the geometric measurements is also straightforwardly derived by assuming an additive Gaussian noise :

$$\begin{bmatrix} \bar{u}_q^m(t) \\ \bar{v}_q^m(t) \end{bmatrix} = \begin{bmatrix} u_q^m(t) \\ v_q^m(t) \end{bmatrix} + \begin{bmatrix} \beta_1^m(t) \\ \beta_2^m(t) \end{bmatrix} \tag{14}$$

where measurement noises $\beta_1^m(t)$ and $\beta_2^m(t)$ are assumed to be Gaussian random vectors of variance σ_β^2. Again, σ_ζ^2 and σ_β^2 are set empirically.

We now define notations related to the data association issue. We call K the set of assignements of measurements to trajectory models, which can be decomposed over time and measurements as follows :

$$K = \Big[K(1), \ldots, K(T)\Big] \tag{15}$$

$$K(t) = \Big\{k_1(t), \ldots, k_{s_t}(t)\Big\} \tag{16}$$

Each assignement variable $k_j(t) \quad (j = s, \ldots, s_t)$ takes values in $[1, \ldots, M]$, thereby indicating to which trajectory model the measurement j is assigned at time instant t.

Let us also introduce Π, the probability of trajectory models, which can also be decomposed over time as follows :

$$\Pi = \Big[\Pi(1), \ldots, \Pi(T)\Big] \tag{17}$$

$$\Pi(t) = \Big\{\pi_1(t), \ldots, \pi_M(t)\Big\} \tag{18}$$

Given a measurement at time t, $\pi_m(t)$ represents the probability that a measurement originates from model m, regardless of which measurement it may be. While K contains binary assignment random variables, the sets $\mathcal{X}$ and Π contain continuous random variables. Classical multi-track extraction methods (JPDAF, MHT) are based on the two following assumptions:

- the assumption that a measurement is associated to one and one trajectory model only, from which the following constraint on assignment variables is inferred :

$$\sum_{m=1}^{M} p\Big(k_j(t) = m\Big) \;=\; \sum_{m=1}^{M} \pi_m(t) = 1 \tag{19}$$

- the assumption that at most one measurement can originate from a moving object at a time. This implies a dependence of assignment variables.

In contrast, the approach we adopt, namely PMHT, relies only on the first of these two assumptions. Consequently, we assume independence of the assignment variables, which allows the factorization of the joint probability of $K(t)$ as described by :

$$p(K(t)) \;=\; \prod_{j=1}^{n_t} p(k_j(t)) \tag{20}$$

It is this very formulation which avoids enumeration of measurement-to-track association hypotheses.

4 Main theoretical aspects of PMHT

4.1 Joint estimation formulation and posterior probability

We recall in this section the main theoretical aspects of PMHT that are used in our method. The search for optimal assignments and states being two interlocking issues, Streit [28] proposed to include the data association problem in the estimation problem; more precisely, to consider the assignment variables as random variables to be estimated along with the state variables. Let us define $\Phi = (\mathcal{X}, \Pi)$. The $\{\pi_m\}_{m=1,\dots,M}$ represent the laws of the discrete variables $k_j(t)$, and estimating Φ according to the *Maximum A Posteriori (MAP)* criterion amounts to a joint estimation of assignments and states. The *a posteriori* distribution can be expressed by :

$$\begin{aligned} p(\Phi|\mathcal{Z}) \propto & p\Big(\mathcal{Z} \mid \mathcal{X}, \Pi\Big) p(\mathcal{X}, \Pi) \\ \propto & \underbrace{\prod_{t=1}^{T} p\Big(Z(t) \mid X(t), \Pi(t)\Big)}_{\text{measurement likelihood}} \qquad \underbrace{p\Big(X(1)\Big) \prod_{t=2}^{T} p\Big(X(t) \mid X(t-1)\Big)}_{\text{prior state evolution}} \end{aligned} \tag{21}$$

Our goal is to find an estimate of Φ which maximizes the posterior probability (21).

Gauvrit and Le Cadre [8] have shown that, in the above expression, the measurement likelihood term can be expressed as the product of conditional likelihoods of measurements $z(t)$, which in turn are defined as a mixture density law, in which the parameters weighing the respective contributions of the elementary laws to the mixture are the prior probabilities of the

trajectory models. This can be written as follows :

$$\prod_{t=1}^{T} p\Big(Z(t) \mid X(t), \Pi(t)\Big) \;=\; (22)$$

$$= \prod_{t=1}^{T} \prod_{j=1}^{s_t} \sum_{m=1}^{M} p\Big(z_j(t) \mid x_m(t)\Big) \pi_m(t) \quad (23)$$

An essential point is that, thanks to the independence assumption between assignment variables, writing (22) as a product of mixture laws (23) is made possible. Direct maximization of (21) is however not feasible, since it is parameterized by the unknown weights $\pi_m(t)$.

Following the work by Redner and Walker [23], the EM algorithm [6] can be used to estimate the parameters of such a mixture density, through an iterative procedure. Let us assume that an initial estimate Φ^0 is available. At the $i+1^{th}$ iteration of the algorithm, in a first step ("E-Expectation" step), an approximation of the *a posteriori* distribution is computed, via its expectation, from measurements and current estimates Φ^i of Φ. In a second step ("M-Maximization" step), a new estimate Φ^{i+1} is computed from the approximation that has just been determined. "E" and "M" steps are alternatively iterated until (guaranteed [6]) convergence. An appropriate and efficient initialization of the recovery problem of multiple trajectories in an image sequence is specified in the next section.

4.2 Association between partial tracks and trajectory models

Spatial proximity or other criteria can supply a short-term temporal link between measurements but, due to the possible lack of detections, in case of occlusion or crossing for instance, this link is sometimes broken. Therefore, our association problem is not more the assignement of the measurements to the trajectory models at each time instant, but the association of available partial tracks to the trajectory models. To this respect, we adapt the method proposed by Giannopoulos et al. [10] for radar and sonar data, and summarize below the main results.

Let us denote $\mathcal{P}$ the set of M partial tracks and $K_l^{\mathcal{P}}$ the assignment of partial track $\mathcal{P}^l$. This assignment takes values in $[1, \ldots, M]$. $\mathcal{P}$ and the set $K^{\mathcal{P}}$ of assignments can be decomposed as follows :

$$\mathcal{P} = \Big\{\mathcal{P}_1, \ldots, \mathcal{P}_M\Big\} \quad (24)$$

$$K^{\mathcal{P}} = \Big\{K_1^{\mathcal{P}}, \ldots, K_M^{\mathcal{P}}\Big\} \quad (25)$$

To apply the EM algorithm, we need to derive the expectation of the logarithm of the *a posteriori* distribution of variables Φ given an estimate Φ_i. This can be expressed as follows, starting from (21) and (23) :

$$Q(\Phi \mid \Phi^i) \;=\; \sum_{m=1}^{M} \sum_{\mathcal{P}_l \in \mathcal{P}} w_{\mathcal{P}_l,m}^{i+1}(t)\; ln[\pi_m(t)] \quad (26)$$

$$+ \sum_{m=1}^{M} \sum_{\mathcal{P}_l \in \mathcal{P}} \sum_{x_j \in \mathcal{P}_l} ln[p(z_j(t) \mid x_m(t))\;]\; w_{\mathcal{P}_l,m}^{i+1}(t)$$

$$+ \sum_{m=1}^{M} ln[p(x_m(1))] + \sum_{m=1}^{M} \sum_{t=2}^{T} ln[p(x_m(t) \mid x_m(t-1))]$$

where $w_{\mathcal{P}_l,m}^{i+1}$ is a weighing factor corresponding to the probability of assigning partial track $\mathcal{P}_l$ to model m, and is defined by :

$$w_{\mathcal{P}_l,m}^{i+1} = \prod_{z_j \in \mathcal{P}_l} \left(\frac{\pi_m^i p(z_j \mid x_m(t))}{\sum_{m=1}^{M} \pi_m^i p(z_j \mid x_m(t))} \right) \tag{27}$$

The maximization of $Q(\Phi \mid \Phi^i)$ can be decomposed into two independent maximizations, first with respect to the parameters of the mixture, the $\pi_m(t)$'s, and second w.r.t. to the states (i.e. the trajectory models), the $x_m(t)$'s. Through these maximizations, one updates the estimate $\Phi^i = (\Pi^i, X^i)$ at iteration $i+1$ to get $\Phi^{i+1} = (\Pi^{i+1}, X^{i+1})$.

The first maximization problem has a simple analytic solution. For every t and m, we get :

$$\pi_m^{i+1}(t) = \frac{1}{s_t} \sum_{j=1}^{s_t} w_{j,m}^{i+1}(t) \tag{28}$$

The second problem consists of the state estimation :

$$\begin{aligned} (x_m(0), \ldots, x_m(T)) \in \\ \underset{Xm}{\operatorname{argmax}} \Bigg\{ & \sum_{\mathcal{P}_l \in \mathcal{P}} \sum_{z_j \in \mathcal{P}_l} ln(\ p(z_j(t) \mid x_m(t))\ )\ w_{j,m}^{i+1}(t) \\ & + ln[p(x_m(1))] \\ & + \sum_{t=2}^{T} ln\,[p(x_m(t) \mid x_m(t-1))] \Bigg\} \end{aligned} \tag{29}$$

In the case of a Markovian process, it is more relevant to maximize the exponential of the expression included in relation (29), that is :

$$p(x_m(1)) \prod_{t=2}^{T} \left\{ p(x_m(t) \mid x_m(t-1)) \prod_{j=1}^{st} p(z_j(t) \mid x_m(t))^{w_{j,m}^{i+1}(t)} \right\} \tag{30}$$

Taking advantage of the Gaussian nature of the measurement noise, this expression can be simplified by introducing a fictitious "synthetic" measurement $\bar{z}_m(t)$ and its covariance matrix $\bar{R}_m$, defined below (relations (32) and (33)). $\mathcal{N}[\bar{z}_m(t), x_m(t), \bar{R}_m]$ denotes the Gaussian probability distribution of variable $\bar{z}_m(t)$, parameterized by its mean $x_m(t)$ and covariance matrix $\bar{R}_m$. At each instant t, we have :

$$\prod_{j=1}^{s_t} p(z_j(t) \mid x_m(t))^{w_{j,m}^{i+1}(t)} \propto \prod_{j=1}^{s_t} \mathcal{N}[z_j(t), x_m(t), (w_{j,m}^{i+1}(t))^{-1} R_m] \propto \mathcal{N}[\bar{z}_m(t), x_m(t), \bar{R}_m] \tag{31}$$

$$\text{with} \qquad \bar{z}_m(t) = \frac{1}{s_t \pi_m^{i+1}(t)} \sum_{j=1}^{s_t} w_{j,m}^{i+1}(t) z_j(t) \tag{32}$$

$$\bar{R} = \frac{R_m}{s_t \pi_m^{i+1}(t)} \tag{33}$$

This transform leads to the classical expression (34) of the *a posteriori* distribution of the state for a *single track* :

$$p(x_m(1)) \prod_{t=2}^{T} \left\{ p(x_m(t) \mid x_m(t-1)) p(\bar{z}_m(t) \mid x_m(t)) \right\} \tag{34}$$

The practical resulting algorithm is particularly simple, since the optimal estimation of $\mathcal{X}$ amounts to M independent estimations using Kalman filtering with smoothing.

5 Initialization stage and tracking algorithm

Let us stress that, in general, the result of the EM algorithm is strongly dependent on the initialization provided for the parameters to be estimated. For our problem, this means that care should be taken to provide the best possible initial guesses for each trajectory model. It is the main purpose of this section to describe the solution we propose to this issue. We expose below how, by utilizing rich information about geometry and velocity of the regions, a meaningful and robust initialization can be elaborated, leading to an original and effective PMHT multiple-object tracking scheme.

Figure 2 includes an overview of the proposed scheme. Since the true number of moving objects, and consequently of trajectories to recover in the image sequence is unknown, we initially set it to M as stated in section 3, where M is the number of partial tracks found within the batch, i.e. in the processed image sequence. The PMHT algorithm requires initializing states and prior probabilities of trajectory models. For the latter, we initially set them in a uniform way, for every instant t and for every model m: $\pi_m^0(t) = 1/M$. Then, the objective is to determine the number of actual trajectories by grouping the partial tracks through the joint trajectory estimation process introduced in section 4.

We exploit the partial tracks to build the M initial trajectories (initial states). Each trajectory model is initially assigned the measurements forming a partial track. We then estimate independently the M models over the whole sequence. Figure 3 illustrates this operation in an example involving three models. A prediction-only estimation mode is used in the Kalman filtering step at time instants when measurements are not available (dashed polygons in fig. 3).

Handling of the geometric component

Tracking of the geometric models by Kalman filters cannot be directly applied by considering that the vertices of the polygonal approximation of the segmentation mask form the measurements of the geometric component. As illustrated in Fig. 4, since polygonal approximations are carried out independently over time, even slightly time-varying segmentation masks may generate significantly different sets of polygonal approximation vertices (regarding the location and the number of these vertices). To solve this issue and supply correct vertices $\tilde{P}_t^j$ for correspondence, we operate as follows (fig. 4) : (1) the predicted polygon and the extracted one are spatially registered with a translation, minimizing the inter-polygon distance defined in [5] with local gradient-descent; (2) for each vertex of the predicted geometric component, the nearest point on the polygon extracted from the image is chosen to be the corresponding measurement.

Let us point out that the prediction/update principle applied to the geometric component by Kalman filtering enables some (limited) degree of non-rigidity in the motion (in addition to the sequence of affine transforms). More precisely, the affine transform assumption is used for the prediction step (use of the global affine motion for all the vertices of a given region), but the adjustement step is carried out locally at each vertex, hence handling, to some extent, articulation and deformation.

1. First estimation of trajectory models $X^0 = (X_1^0, \ldots, X_M^0)$ from partial tracks.

2. Detection and elimination of measurements corresponding to occlusion /disocclusion phases, if any.

3. Re-estimation of trajectory models $X^0 = (X_1^0, \ldots, X_M^0)$, having discarded perturbed measurements in step 2. These models serve as the initialization for the EM algorithm.

4. Initialization of the mixture model parameters

$$\Pi^0 = (\pi_1^0, \ldots, \pi_M^0), i = 0.$$

5. For each partial track and for each trajectory model, compute :

- the probability of association of the partial track to the model :

$$w_{\mathcal{P}_l,m}^{i+1} = \prod_{j \in \mathcal{P}_l} \left(\frac{\pi_m^i p(z_j \mid x_m(t))}{\sum_{m=1}^{M} \pi_m^i p(z_j \mid x_m(t))} \right)$$

- the prior probability of each trajectory model :

$$\pi_m^{i+1}(t) = \frac{1}{m_t} \sum_{j=1}^{m_t} w_{j,m}^{i+1}(t)$$

6. For each trajectory model and each time instant, compute the *a posteriori* measurement and its covariance matrix :

$$\begin{aligned} \bar{z}_m(t) &= \frac{1}{s_t \pi_m^{i+1}(t)} \sum_{j=1}^{s_t} w_{j,m}^{i+1}(t) z_j(t) \\ \bar{R}_m &= \frac{R_m}{s_t \pi_m^{i+1}(t)} \end{aligned}$$

7. For each trajectory model, estimate the state at each time instant by Kalman filtering with smoothing, considering the *a posteriori* measurements and their covariance matrices computed in step **6.**

8. Increment $i = i + 1$ and go to step 5 if the stopping criterion (35) is not met.

9. Decision on possible association of several partial tracks.

Figure 2: Overview of the proposed scheme.

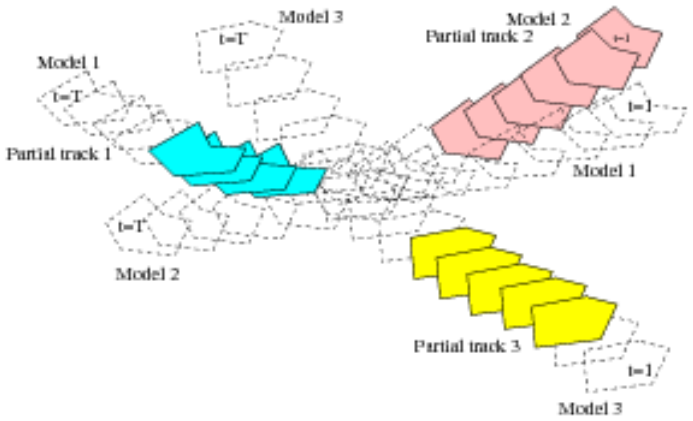


Figure 3: Building initial states, in the case of three partial trajectories (only the geometric component is shown here). Dashed lines represent temporal extensions, when a prediction-only mode is employed.

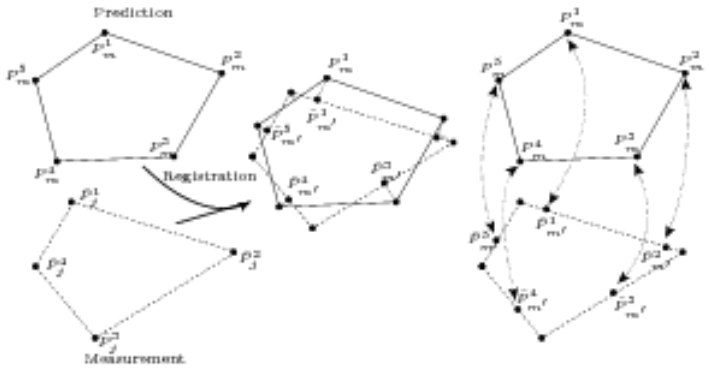


Figure 4: The two polygons, one corresponding to the prediction computed from the current region trajectory model and the other to the extracted region, are first registered using a translation. Then, for each vertex on the model, the closest point on the measurement polygon is considered, so as to attempt to obtain pairs of points that approximately correspond to the same physical point. For the sake of figure clarity, the predicted geometric model and the polygonal silhouette of the extracted region are drawn far apart.

Discarding perturbed measurements

We noticed that the reliability in the "prediction-only" mode of the state is strongly dependent on the accuracy of the last few measurements before the filter switches to this mode. Typically, these last few measurements can correspond to a progressive occlusion phase (Fig .1). Such an issue arises both for progressive appearance and disappearance of a region. The geometric component is particularly affected, since the extracted region and its measured silhouette reveal only the visible part of the object. Therefore, we decided to discard such "uncertain" measurements. We carry out detection of occlusion and disocclusion phases according to the criterion introduced in [15], since it has proved effective enough. In short, it consists in detecting unexpected strong temporal variation of the area of the tracked region support. We predict the area of this region from time t to time $t+1$, using the divergent component of the 2D motion field of the region (due to object motion towards or away from the camera, or camera motion). It can be straightforwardly computed from the 2D affine motion model (given by $\frac{1}{2}(a_2+a_5)$) estimated over the considered region at time instant t. We then examine an "innovation" variable, which is the difference between area of the segmented region at time t, and its prediction. Temporal upward or downward jumps of this variable are then detected using Hinkley's test. Besides its simplicity, the interest of this test is two-fold. Since it is cumulative over time, it can detect (dis)occlusion phases with various speed with the same threshold. It also provides conveniently the time at which the (dis)occulsion phase starts (which is by construction a little earlier than the time at which it is detected). Once the (dis)occulsion phases have been identified, if any, the corresponding measurements are discarded, and the states of all models are re-estimated over the batch.

Iteration and convergence of the EM algorithm

From these initial state estimates and prior model probabilities, the two steps of the EM algorithm are iterated : computation of the measurement-to-model assignment probabilities given the current states, derivation of prior probabilities of models and of the "synthetic" measurements $\bar{z}_m(t)$, estimation of the states over the batch. Convergence is considered obtained when the following condition is met:

$$\max_{j,m,t} \mid w^i_{j,m}(t) - w^{i-1}_{j,m}(t) \mid < \delta_w \tag{35}$$

The parameter δ_w is typically set to 0.001.

The key parameters of the algorithm that the user should set are the process and measurement noises. Automatic learning of appropriate values from image sequences are beyond the scope of this paper, notably because their setting should exploit application-dependent knowledge, or extensive training data.

Convergence of the EM algorithm leads to an optimal (in the sense defined of relation (21)), stable, assignment of measurements to trajectory models. A policy to recover the *full tracks*, in other words to associate *partial tracks*, can be defined on the basis of the values obtained for these assignments $w^{i+1}_{p_1,m}$. In practical experiments, we observe that a clear convergence of $w^{i+1}_{p_1,m}$'s to 1 or 0 occurs in most cases, respectively if two partial tracks should intuitively clearly be associated or not. Simple thresholding below e.g. 10e-3 or above 1-10e-3 easily identifies such situation. On the other side, typical ambiguous cases include :

- two partial tracks which trajectories are not clearly the continuation of one another, but might be (this may occur in the presence of temporary occlusions) ;

- two partial tracks overlapping in time, that both are in plausible continuity of a third partial track, that occurs earlier or later.

In the first case, weights take intermediate values between 0 and 1. In the second case, the weights associating the third partial track to the two trajectory models arising initially from the two plausible matching partial tracks are typically close to 0.5, since these weights should sum to 1. Existence of such configurations may be identified.

A practical rule, in the context of region tracking, is suggested by our experiments. In [15], two trajectory models are to be grouped if, over a sufficient time interval, they are consistent both in position and velocity. In contrast, we suggest to only demand consistency in position, and leave more flexibility on the evolution of the kinematics during occlusion phases. Besides, the influence of kinematics remains via the state equation (12). Moreover, we globally handle the determination of multiple trajectories, whereas in [15], the problem is stated by considering each trajectory individually.

More generally, the probabilistic nature of the results provided by our technique opens interesting perspectives for variations in the decision-taking phase. The present paper proposes a technique for *inferring* the association probabilities. From there, one may introduce some cost associated to each type of error, depending on the application, and apply various decision strategies (Bayesian, minimax,...) to conclude. Finally, formalisms that penalize overall complexity in explaining the scene may be introduced to supply automatically an interpretation of the scene, by trading trajectory continuity for global scene simplicity.

6 Experimental results

We report experimental results for two real image sequences involving complex situations. The first one is the "Breakfast" sequence, acquired in our lab and which was already described in Section 1 (Fig.1). The scene comprises four partial tracks : two per object, as each object undergoes temporary total occlusion. Then, four trajectory models are initially created and estimated. At convergence, finally two global trajectories are retained and estimated. For this sequence, initial and final estimated trajectory models are respectively plotted on Fig. 5a and 5b, with measurements. It can be noticed that, at convergence of our algorithm, the four partial tracks are correctly grouped in two pairs, despite the relatively complex crossing situation. Only the gravity centers of the geometric models are indicated for clarity sake.

Fig. 6a and 6b respectively show the computed geometric measurements, and the estimated geometric models at convergence, superimposed over the first image of the sequence. The algorithm supplies relevant geometric models, including the whole silhouette of the regions at instants when partial or total occlusions take place. Convergence is obtained in about 20 iterations for this sequence.

As an example, a result for the kinematic model is provided in Fig. 7, for the translational parameter a_1 of the motion model. Measurements and estimated values of a_1 are plotted for two trajectory models corresponding to two partial tracks in the "Breakfast" sequence, that should be associated. They are provided at initialization (Fig 7a,b) and at convergence (Fig 7c,d) of the EM algorithm. The (conservative) prediction-only mode employed for estimating the kinematic model when no measurement is available consists in keeping the last filtered value available constant. The need for this switching of evolution model arises from the following observation : the last few measurements before switching to prediction-only mode (e.g. corresponding to a occlusion) are not reliable enough to allow long-term in prediction-only mode based on a higher-order evolution model on motion parameters, so this simpler model is only employed in

this context. As the two partial tracks are correctly associated at convergence, it appears that the state estimation corresponds to Kalman smoothing.

The second sequence depicts an outdoor scene. The "Van" sequence is a crossroads scene (a few images of the sequence are displayed in Fig 8a), in which the white vehicule (partial track 2) crosses (behind) a van (partial track 1), and reappears on its left (partial track 3). Fig 8b shows the corresponding motion-based segmentation maps. The dark car closely following the van is not differenciated by the motion-segmentation scheme from the van it is following, as their motions are very similar. Due to the short-term linkage provided by the motion segmentation algorithm, three partial tracks and associated object trajectory models are generated for the sequence, two of which actually correspond to the same white vehicule. Values of the kinematic measurements and estimated motion models, exemplified by a_1, are provided in Fig. $8c_1$ and $8c_2$ respectively at initialization and at convergence of the EM algorithm. It can be observed that model 2 fits partial track 3, while model 3 mismatches partial track 2. As explained in the previous section, we state that a one-direction fit suffices to associate the two partial tracks at hand. The evolution of the association weights $w_{\mathcal{P}_i,m}$ over iterations is supplied, for trajectory models 2 and 3 with partial track 3, in Fig $8c_3$. Hence, our tracking method was able to correctly decide that there were only two relevant different entities (i.e., $\mathcal{M} = 2$), and to accurately recover the corresponding two entire trajectories, despite the first partial, then total occlusion, and the crossing situation.

The running time of the technique on a 60-image batch is about 2 seconds (C++ implementation) for the data association part, which is the contribution of this paper. The processing time required by prior motion segmentation from the image sequence is about an order of magnitude higher.

The MHT technique is based on the NP-complete enumeration of association hypotheses, usually requiring application of pruning techniques to the hypotheses tree. In the PMHT technique, computational complexity only grows moderately with the number of partial tracks. The examples considered here only involve a few regions and computational cost should be low both for MHT and PMHT. In general, however, PMHT posseses three advantages for the region-tracking problem:

- The more computationally-expensive features are added to the regions (e.g. the geometric features, included in this paper; color distribution, as a valuable extension), the greater the computational advantage of PMHT over hypothesis enumeration. Besides, introduction of pruning/gating techniques for MHT would require ad-hoc tuning for each feature.
- The context chosen was that of a availability of a short-term link between regions. In situations where this link does not exist, the combinatorial issue is strong even for sequences such as the ones presented in the paper.
- Besides combinatorial issue, there is an intrinsic advantage in probabilistic modelling of the associations, in that it takes naturally into account uncertainties on measurements and models, and also provides confidence evaluation as an output and hence enabling various decision-taking policies.

7 Conclusion

We have presented an original and efficient method for tracking multiple objects in an image sequence. It involves the association of partial tracks of regions, while jointly estimating the

trajectories of these regions. We have introduced the modelling of geometric and kinematic components of regions in the PMHT framework. From an adequate model initialization scheme, an iterative EM procedure leads to a stable configuration of trajectory models from which associations can be inferred and entire trajectories of the physical moving objects recovered. The proposed tracking method has been validated by experiments on real image sequences involving complex events such as partial occlusion, total (temporary) occlusion and crossing.

The practical interest of the proposed method is several fold. The understanding of the sequence content is improved and a rich description of the content is provided: region motions and trajectories with the whole silhouette of objects are estimated over the whole sequence, including when measurements are either not available, or not reliable. A possible major improvement on the performance of the scheme could be obtained by adding intensity or color related descriptors to the measurements, and modelling their temporal evolution, as for instance described in [11].

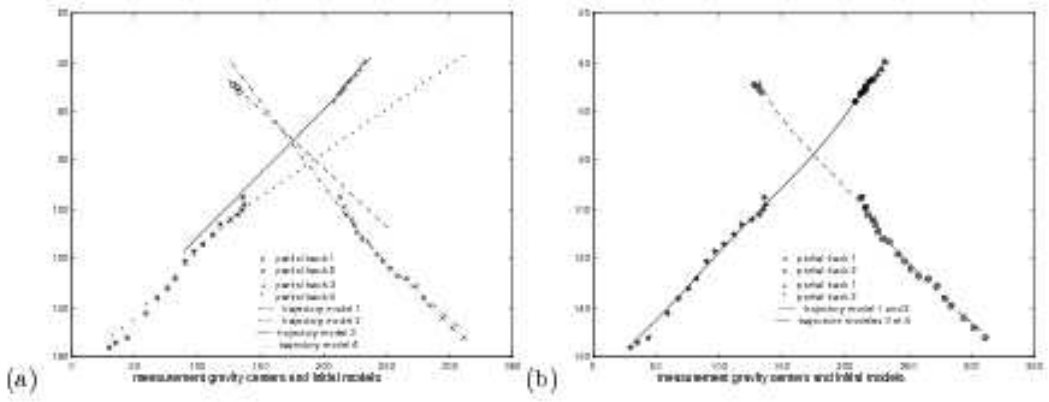


Figure 5: "Breakfast" sequence : measurements and four initially estimated partial trajectories (a) and the two finally estimated global trajectories at convergence (b). Only the gravity centers of the geometric models are displayed.

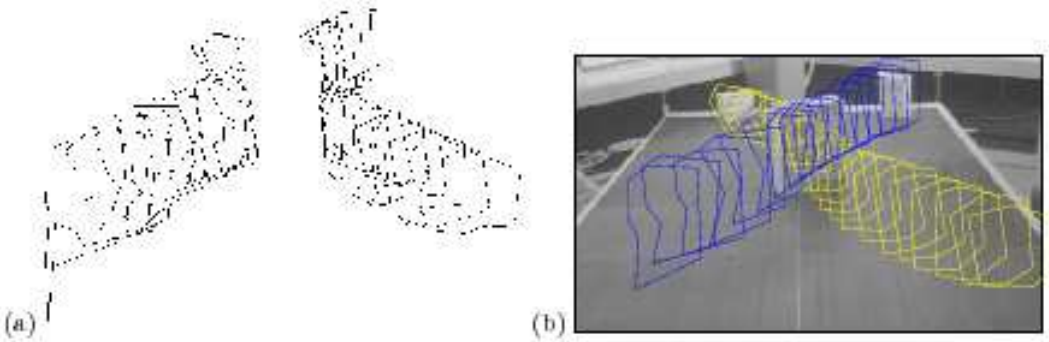


Figure 6: "Breakfast" sequence : measured polygonal silhouettes (a), estimated geometric models at convergence, superimposed on the original image at $t = 0$ (b). For the sake of clarity, only one out of two geometric models (in time) are represented.

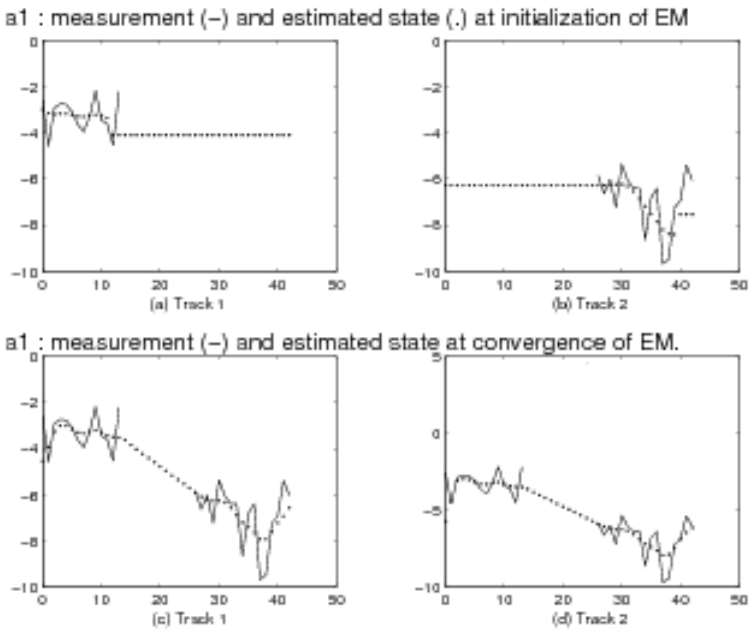


Figure 7: "Breakfast" sequence : estimated (filtered) values (dotted line) of parameter a_1 (kinematic component) for two of the four trajectory models, plotted at initialization (a,b) and at convergence (c,d) of the EM algorithm.

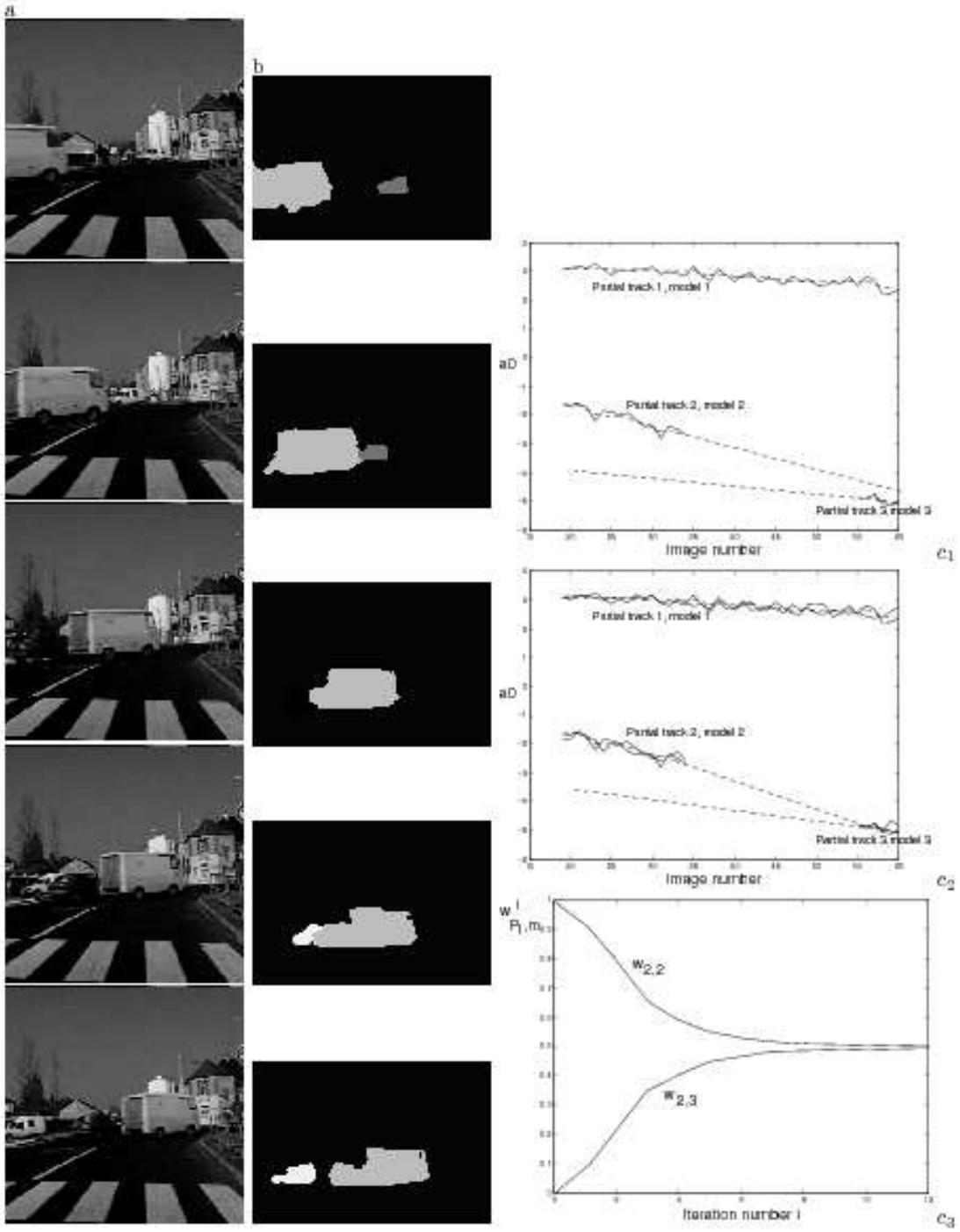


Figure 8: Column (a): images from the "Van" sequence, at time instants $t = 19, 31, 47, 55, 61$. Column (b): obtained motion segmentation maps for these images. Column (c): evolution over the sequence of the affine motion parameter a_1 for the three models and three partial tracks, at initialization (c_1) and at convergence of the EM algorithm (c_2), evolution over the iterations of association weights $w^l_{P_1,m}$, for $l = 2$, $m = 2$ and $m = 3$.

References

[1] Y. Bar-Shalom and X.R. Li. *Estimation and Tracking : Principles, Techniques and Software*. Artech House, Boston, 1993.

[2] A. Blake and M. Isard. *Active contours*. Springer, 1998.

[3] I.J. Cox. A review of statistical data association techniques for motion correspondance. *Int. Journal of Computer Vision*, 10(1):53–66, 1993.

[4] I.J. Cox and S.L. Hingorani. An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(2):138–150, February 1996.

[5] P. Cox, H. Maître, M. Minoux, and C. Ribeiro. Optimal matching of convex polygons. *Pattern Recognition Letters*, (9):327–334, June 1989.

[6] A.P. Dempster, N.M Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society Ser. B*, 39:1–38, 1977.

[7] T.E. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Research*, pages 173–184, July 1983.

[8] H. Gauvrit, C. Jauffret, and J.P. Le Cadre. A formulation of multitarget tracking as an incomplete data problem. *IEEE Trans. on Aerospace and Electronic Systems*, 33(4):1242–1257, October 1997.

[9] M. Gelgon, P. Bouthemy, and J-P. Le Cadre. Associating and estimating trajectories of multiple moving regions with a probabilistic multi-hypothesis tracking approach. In *Proceedings of Int. Symposium of Physics in Image Processing*, pages 80–83, Paris, January 1999.

[10] E. Giannopoulos, R. Streit, and P. Swaszek. Multi-target track segment bearing-only association and ranging. In *31rst Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, November 1997.

[11] R. Hammoud and R. Mohr. Mixture densities for video objects recognition. In *International Conference on Pattern Recognition (ICPR'2000)*, pages 71–75, Barcelona, Spain, September 2000.

[12] C. Hue, J-P. Le Cadre, and P. Pérez. Tracking multiple objects with particle filtering. *IEEE Trans. on Aerospace and Electronic Systems*, 38(3):791–812, July 2002.

[13] M. Isard and A. Blake. CONDENSATION - conditional density propagation for visual tracking. *Int. Journal of Computer Vision*, 1(29):5–28, 1998.

[14] F. Marques and C. Molina. Object tracking for content-based functionalities. In *SPIE Visual Communication and Image Processing (VCIP-97)*, volume 3024, pages 190–198, San Jose, 1997.

[15] F. Meyer and P. Bouthemy. Exploiting the temporal coherence of motion for linking partial spatio-temporal trajectories. In *Proc of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 746–747, New-York, June 1993.

[16] F. Meyer and P. Bouthemy. Region-based tracking using affine motion models in long image sequences. *CVGIP : Image Understanding*, 60(2):119–140, September 1994.

[17] K.J. Molnar and J.W Modestino. Application of the EM algorithm for the multitarget/multisensor tracking problem. *Signal Processing*, 46(1):115–128, January 1998.

[18] F. Moscheni, S. Bhattacharjee, and M. Kunt. Spatiotemporal segmentation based on region merging. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(9):897–915, September 1998.

[19] J-M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Jal of Visual Communication and Image Representation*, 6(4):348–365, December 1995.

[20] J.M. Odobez and P. Bouthemy. Direct incremental model-based image motion segmentation for video analysis. *Signal Processing*, 66(3):143–156, May 1998.

[21] N. Paragios and R. Deriche. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:266–280, March 2000.

[22] C. Rasmussen and G.D. Hager. Joint probabilistic techniques for tracking multi-part objects. In *Proc. of Int. Conf. on Computer Vision and Pattern Recognition*, pages 18–26, Santa-Barbara, June 1998.

[23] R.A. Redner and H.F Walker. Mixture densities, maximum likelihood and the EM algorithm. *Society for Industrial and Applied Mathematics - SIAM Review*, 26(2):195–239, 1984.

[24] D.B Reid. An algorithm for tracking multiple targets. *IEEE Trans. on Automatic Control*, 24(6):843–854, December 1979.

[25] M. Ringer and J. Lasenby. Multiple hypothesis tracking for automatic optical motion capture. In *Proc. of European Conference on Computer Vision (ECCV'2002)*, pages 524–536, Copenhaguen, Denmark., May 2002.

[26] A. Strandlie and J. Zerubia. Particle tracking with iterated Kalman filters and smoothers: the PMHT algorithm. *Computer Physics Communications*, 123(1-3):77–86, 1999.

[27] R.L. Streit. *Studies in Probabilistic Multi-Hypothesis Tracking and Related Topics*, volume SES 98-01. Naval Underwater Warfare Center Division, February 1998.

[28] R.L. Streit and T.E. Luginbuhl. A probabilistic multi-hypothesis tracking algorithm without enumeration and pruning. In *Proc. of the 6th Joint Service Data Fusion Symposium*, pages 1015–1024. Laurel, June 1993.

[29] J-P. Tarel, S-S. Ieng, and P. Charbonnier. Using robust estimation algorithms for tracking explicit curves. In *Proc. of European Conference on Computer Vision (ECCV'2002)*, pages 492–507, Copenhaguen, May 2002.

[30] K. Wall and P.E. Danielsson. A fast sequential method for polygonal approximation of digitized curves. *Computer Vision, Graphics, and Image Processing*, (28):220–227, 1984.

[31] M.A. Zaveri, U.B. Desai, and S.N. Merchant. Pmht based multiple point targets tracking using multiple models in infrared image sequence. In *Proc. of IEEE Conf. on Advanced Video and Signal Based Surveillance (AVSS'03)*, pages 73–79, Miami, USA, July 2003.

[32] Z. Zhang and O Faugeras. Three-dimensional motion computation and object segmentation in a long sequence of stereo frames. *Int. Journal of Computer Vision*, 7(3):211–241, 1992.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
9 October 2003 (09.10.2003)

PCT

(10) International Publication Number
WO 03/083716 A1

(51) International Patent Classification[7]: G06F 17/30

(21) International Application Number: PCT/FI02/00277

(22) International Filing Date: 28 March 2002 (28.03.2002)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: **NOKIA CORPORATION** [FI/FI]; Keilalahdentie 4, FIN-02150 Espoo (FI).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **MYKA, Andreas** [DE/FI]; Ida Aalbergintie 3 C 31, FIN-00400 Helsinki (FI). **YRJÄNÄINEN, Jukka** [FI/FI]; Ollinmäentie 5, FIN-33480 Ylöjärvi (FI). **GELGON, Marc** [FR/FR]; 4 rue des reinettes, 44300 Nantes, France (FR).

(74) Agent: **RUUSKANEN, Juha-Pekka**; Page, White & Farrer, Runeberginkatu 5, 10th Floor, FIN-00100 Helsinki (FI).

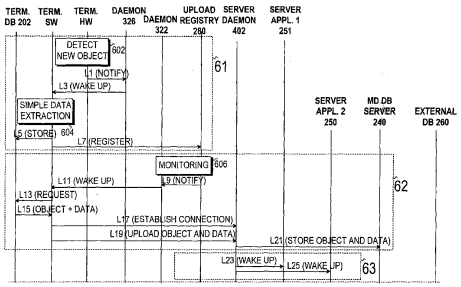
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) **Title:** ENHANCED STORING OF PERSONAL CONTENT



(57) **Abstract:** The invention relates generally to the access to and the creation of content in the context of a mobile communications system and more specifically to archiving personal content of mobile users and for providing this content to mobile users in the most flexible and personalized ways. The core of the invention is how personal content acquired by the user may be further enhanced and stored in a safe box-like remote repository for future purposes. At least one remote data repository is assigned for the use of mobile terminals. Personal content is acquired and stored in the mobile terminal. Selected personal content is then transferred between the storage means and the remote data repository through said telecommunications system, the means to include predetermined criteria, the fulfillment of which initiates said transfer.

WO 03/083716 A1

ENHANCED STORING OF PERSONAL CONTENT

FIELD OF THE INVENTION

The invention relates generally to access to and the creation of content in the context of a mobile communications system. More specifically, the invention relates to a method and system for archiving the personal content of mobile users and for providing this content to mobile users in the most flexible and personalized ways. The content refers here to any multimedia data, including e-mails, text messages, images, audio files, calendar entries, log information, and e-commerce data. The invention relates to acquiring personal content on a mobile terminal, storing it in a remote repository, and retrieving it from the remote repository.

BACKGROUND OF THE INVENTION

The strong growth in the number of Internet users and services provided through the Internet has been one of the most remarkable phenomena in communications in recent years. Another current trend is the strongly increasing use of various mobile terminals, such as laptops, PDA (Personal Digital Assistant) equipment, and intelligent telephones.

These two rapidly evolving network technologies, wireless communication and the Internet, are gradually converging. So far this converging development has been progressing rather slowly, since most of the technology developed for the Internet has been designed for desktop computers and medium or high bandwidth data connections. It has, therefore, been difficult to introduce the IP-based (IP = Internet Protocol) packet services to the mobile environment, which is characterized by less bandwidth and poorer connection stability in comparison to fixed networks, and where the terminals have many fundamental limitations, such as smaller displays, less memory, and less powerful CPUs, as compared to fixed terminals. However, the development of IP-based packet services for the mobile environment will occur at an increasing rate in the foreseeable future. This is partly due to the demand created by the market and partly due to the evolvement of new technologies designed to meet the various requirements of mobile networks, such as sufficient quality of service and data security. The increasing market demand is based on the rapid increase in the

popularity of the Internet: Internet users are often also mobile subscribers and thus may also want to use in their mobile terminals the services familiar to them from the Internet environment. This commercial demand in turn enables investments necessary for the development of mobile services. The said new technologies include GPRS (General Packet Radio Service) and WAP (Wireless Application Protocol), for example. GPRS aims at providing high-quality services for GSM subscribers by efficiently utilizing the GSM infrastructure and protocols. WAP, in turn, defines a set of standard components enabling communication between mobile terminals and servers providing service in the network. WAP utilizes proxies that connect the wireless domain with the WWW domain.

The above-described development will in the near future turn the mobile terminals into versatile multimedia tools. In addition to the features that current mobile terminals include, these future terminals will have a variety of sensors for multimedia data, for example, such as a camera and a location sensor. Besides the technical feasibility of constructing such devices, it is important that the users get a clear benefit from using such terminals and that the telecommunications system to which the terminals belong does not pose restrictions on the efficient use of the devices.

In comparison to already existing multimedia tools, such as digital cameras, the recent development of mobile terminals can offer a variety of new multimedia related services, as the technological solutions used by the mobile terminals and the mobile network infrastructure enable various possibilities not seen before. On the other hand, the interconnected networks, such as the Internet, act as enabling factors as well. The possibilities thus created have so far mostly been unexplored, leaving space for innovative practices and new service models within the communications industry.

One example of the immense possibilities mentioned above is sometimes referred to with the general term metadata. Metadata itself is data about data, defining new relations inside a batch of data, or building new ontological layers. The existing solutions to deploy metadata are still far away from effectively utilizing all the possibilities offered via mobile terminals. Some prior art examples to refer to here are described in more detail in U.S. Patent 6,282,362 and European patent application 1 004 967. Typically, images are an important type of multimedia information, and metadata may

indicate the position where an image was taken or information describing the subject of an image.

Some idea of prior art services necessary for the future mobile environment can in principle be found from the international publication WO 00/57315 and U.S. Patent 6,105,042. The idea of eliminating limitations posed by the finite terminal memory and low bandwidth connection between the mobile terminal and the mobile network implies some straightforward solutions for a few typical cases, which are presented in the references.

However, none of the solutions referred to is capable of offering a total solution for a mobile terminal user with respect to the flexibility of storing, transfering, and using the personal content acquired by the user or the terminal. Because all possible solutions have been developed from a narrow point of view and aimed for resolving a single problem at a time, the demands raised by the users, as well as the possibilities offered from the versatility of the systems used, have largely not yet been met.

The objective of the invention is to introduce a novel concept for providing the users with an enhanced method and system for storing personal content. The dependent claims describe some aspects of the invention.

SUMMARY OF THE INVENTION

The objective of the invention is to accomplish a solution which allows efficient and user-friendly mechanisms for providing personalized services to the mobile users in the context of their personal data. This objective is achieved with the solution defined in the independent patent claims. The core of the invention is the mechanism how personal content acquired by the user may be further enhanced and stored in a safe box-like remote repository for future purposes.

According to the invention, access to stored objects can be provided for mobile users in the following manner. First, at least one remote data repository is assigned for the use of each of the terminals, the repository being operationally connected to a telecommunications network for storing personal content. Personal content is acquired by the mobile terminal, which has been adapted to be in wireless communication with a telecommunications network. The personal content acquired is stored in the mobile terminal and then selected personal content is transferred between the storage

means and the remote data repository through said telecommunications system, the means including predetermined criteria, the fulfillment of which initiates said transfer.

Stored personal content is accessed from the mobile terminal by i) requesting an object including stored personal content from the mobile terminal, ii) receiving a predetermined return code if the requested object is not located in the mobile terminal, and iii) further requesting the object from the remote data repository if the return code indicates that the requested object is not located in said storage means.

In accordance with one aspect of the present invention, a server is connected to said remote data repository for managing objects and information extracted and/or generated based on said objects, the objects and information to include the personal content stored in the remote data repository.

In accordance with one aspect of the present invention, said information related to said object is updated to indicate that the object has been requested by the mobile terminal. Then the updated information is stored in the remote data repository.

In accordance with one aspect of the present invention, a register is updated, the register to include objects and/or extracted data stored at least at one point in time in the mobile terminal storage means. In accordance with another aspect of the present invention, this may include marking deleted and/or transferred objects and/or extracted data that has been transferred to the remote data repository.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is described more closely with reference to FIG. **2-15** of the accompanying drawings, in which

FIG. **1** illustrates a mobile network wherein prior art services are provided to the users,

FIG. **2** is a schematic diagram of a system which can be used to provide enhanced data storage capabilities according to the invention, the diagram describing network elements favorable when implementing some embodiments of the invention,

FIG. **3A** shows sample content of a software block of a user terminal **100** which can perform some tasks described in a preferred embodiment,

FIG. **3B** illustrates the hardware block of a user terminal **100**,

FIG. **3C** illustrates the contents of the storage means of a user terminal **100**,

FIG. **4A** illustrates the functional blocks of the software in the MD DB server **240**,

FIG. **4B** represents the contents of an exemplary remote data repository **242**,

FIG. **5** is an example of the functional blocks of the upload registry **280**,

FIG. **6** is a diagram showing examples of tasks performed in the terminal prior to the transfering of data, messaging related to the transfer of the content to the remote data repository, waking service applications, deleting transferred data, and generating data at least partially based on the transferred content,

FIG. **7** is a diagram of exemplary terminal hardware which is detecting new objects,

FIG. **8** is a diagram of an exemplary terminal daemon **326** which wakes up the right terminal application,

FIG. **9** is a diagram of an exemplary terminal application which can extract data from acquired objects and register objects and data ready for transfer,

FIG. **10** is a diagram showing a possible operation model of an upload registry **280** run by the network operator, for example,

FIG. **11** is a diagram of an exemplary reachable terminal daemon **322** in the network which can initiate applications when requested from the network,

FIG. **12** is a diagram showing a possible operation model of MD application **334** which can, for example, take care of uploading the stored personal content into the remote data repository,

FIG. **13** is a diagram showing an exemplary operation of an MD DB server **240**,

FIG. **14** is a diagram showing exemplary operation of an application server **250**,

FIG. **15** is a diagram showing exemplary operation of a deletion application **324** run in the mobile terminal.

DETAILED DESCRIPTION OF THE INVENTION

FIG. **1** shows a schematic diagram of a prior art mobile network **110** connected to a communications network **140** such as the Internet via a gateway element **130**. These kinds of arrangements are widely used to provide services to user terminals, of which one mobile terminal **100** is illustrated in the figure. Mainly, the terminals are in connection with the mobile network via base transceiver stations **120**, which in plurality comprise the radio access network of the network **110**.

Many services which are provided to the users are produced in different servers **150**, an example of which is a WWW/WAP server illustrated in FIG. **1**. The servers **150** are mostly connected directly to the Internet and provide many different services, such as following the stock exchange rates in accordance with criteria supplied by the user subscribing to the service in question. When the server detects that some criterium is fulfilled, it notifies the user by sending a message. Further, services such as directory services or anonymous chat services may be implemented using a server system similar to the example above.

FIG. **2** shows some aspects favorable for designing a network architecture in view of the recent development trends. Because the mobile terminals have been evolving towards versatile multimedia tools, they are supplied with some applications. Examples of typical applications are, for example, camera user interface and data storage logic. The application data which forms the personal content is stored in a local database **202**, which can in practice be a memory chip, a local disk drive, or something else providing the user with reliable means for storing information. According to the invention, the mobile terminal **100** is provided with a plurality of applications, of which two examples **200** and **201** are presented in FIG. **2**. The applications have means to access this content and, if necessary, to perform simple analysis tasks or to transfer the content to a remote data repository **242**. Further, the system comprises an MD DB server **240**, which corresponds to a Media Diary Database server. The MD DB server has several functionalities, and it not only controls access to the remote data repository but performs other tasks very similar to how a user uses a traditional diary and notebook.

The Media Diary (MD) system is the multimedia equivalent to the traditional server. Its parts may include an MD DB server **240**, an MD application **334** in the mobile terminal, various applications, and some other parts described below in more detail. The different system parts are intended to work together so that the strengths of each component may be used in the best way. The MD DB server does not need to be an extraordinary server; a commonly used server will do. The definition MD DB is more inclined to description of the purpose for using the server to access a database. Thus the database corresponds to the MD DB, which is a remote repository where objects of personal content may be stored, for example.

Further, the system comprises a plurality of different application servers **250** and **251**. It is to be noted that the servers need not be separate elements, but that in some cases the applications may be stored in the MD DB server as well. The same applies to the remote data repository **242**, which can be included in the MD DB server. According to one aspect of the present invention, the MD DB system includes a server, a data repository, and means to execute some applications stored somewhere in the network. The purpose of the MD DB system is, on the one hand, to provide the user with reliable data storage, and on the other hand, with a possibility to easily gain the advantages of personalized services.

The system has access, if necessary, to an external database **250**. This can easily be implemented using the Internet or some other communications network.

In order to archive this data, the user data can be transferred from the restricted and expensive local data storage **202** to the bigger and cheaper remote data repository. Data that has been archived can then be (temporarily) removed from the local data storage **202** and the valuable storage space can be reused for data that is to be considered more needed at a specific time. For the uploading task an upload registry **280** may be involved. An example of a possible structure of the upload registry is described in more detail below, but the upload registry may include some of the following: Reading and/or receiving identifiers identifying data to be uploaded, or mobile terminal identification information (such as the phone number or IP address of the terminal) having items to be uploaded. The upload registry may monitor the status of the register and check when a predefined criteria is met, such as when the gross size of the data to be transferred exceeds some limit,

or the transfer price per data unit drops below a predefined threshold, and so on.

The upload registry may also include means for receiving personal content to be transferred to the remote data repository, or means for sending a request (or for waking up a terminal application) to make a connection either to the upload registry or to the MD server or even directly to the MD DB.

Typically, modern mobile networks also include other practical means, such as a user positioning system **282** and a billing system **284**. Some components such as the positioning system **282** or the external database **260** are already known from prior art, but they are presented here only with reference to FIG. **2** because some aspects of the present invention enhance the use of these systems in the manner described in the patent claims.

FIG. **3A** shows a schematic diagram of a software block of the user terminal **100**. An application **200** which can be used to extract data from an object includes definitions **302** which define some settings, such as on which kind of objects the application is capable of working, then possibly some adjustable parameters, such as what system (coordinate system, symbolic/address system, etc.) is used for location information of the user and so on. An object selection block **304** may be used to select objects on which the extraction block **306** is to perform data extraction and so forth. Application **201** comprises definitions **312** similar to definitions **302**, as well as the analysis block **314** and selection block **316**.

The mobile terminal may comprise a plurality of other applications **330** as well, in addition to or instead of applications **200** and **201**. The mobile terminal usually has some form of user interface UI block **332**. The role of the UI block is to provide the user with a convenient way to set his/her preferences and to monitor the operation of the terminal software SW and hardware HW, and so forth. According to one aspect of the present invention, the user terminal may have a Media Diary MD **334** application as well. The MD application may even correspond to a sort of operating system, such so that via the UI block the MD application rules the applications **200**, **201**, **330** and so on. Typically, some mobile terminals also comprise a browser **328**, which is one solution for getting the needed updates for the MD application and other applications.

The MD application **334** is intended to convert the user terminal into a versatile multimedia tool capable of providing special services related to the personal content acquired by the user. Such services are various, but in view of the enhanced data storage functionality, basically the services handle the association of metadata related to personal content or data which has been extracted from said personal content, to the extraction of information from said content, to the transference of the content to and from the remote data repository, to the accessing of said stored content, and to performing some actions like deleting obsolete or outdated information from the user terminal and so forth. In principle, one goal of the MD application is to provide the user with a user interface and means to set-up all the definitions and operating models related to these functionalities, thus acting as a sort of front end. Even if the tasks described above are performed by dedicated program applications, some of which reside in the mobile terminal and some in a networked computer or server, and even if they are adapted to be independent in the sense that this special MD application is not absolutely necessary, it is under current development work in order to offer the user a single point of control and use.

Further, the user terminal has two different daemons available. The network reachable daemon **322** takes care of connections initiated from the mobile network **110** or some other communications network **140**, such as the Internet. The daemon for internal application **326** acts as a middleman between the hardware and software. It may also monitor the action of other applications and perform some predetermined tasks when it considers them necessary.

The reasoning behind at least one daemon with this kind of multifunctional property will be explained in more detail with reference to FIG. **6**, **8**, and **11**, and similarly, the reasoning behind the different applications stored in the mobile terminal is discussed below with reference to FIG. **6**, **9**, **12**, and **15**.

FIG. **3B** is a schematic block diagram of the hardware block of the user terminal. In this context the hardware is considered functionally different from the storage means **202**, but it is to be understood that it is also possible to implement both functionalities together in the hardware part, basically because the physical realization of the storage means always requires some sort of hardware. The hardware block has a database accession block **362**

with means to perform operations on the storage means **202**. Then the hardware block has means in the mobile network communication block **364** to be in communication with the mobile network **110** and with its base transceiver stations **120**. Further, the object generation block **366** can assist in generating personal content objects, generated using or via some part of the hardware be they digital images, calendar entries, speech or text messages,. The system control block **368** supervises the system and keeps the different functional blocks in the hardware running.

FIG. **3C** is a simplified block diagram of the local storage means **202** of the mobile terminal. First, the storage means have an object register **380** which is intended for storing personal content. As discussed below, the object register indicates, on one hand, the objects which are available locally and the objects which have to be retrieved from the remote data repository, on the other hand. In response to a query requesting an object, the local storage means returns a code indicating that the object is not available locally and has to be fetched from the remote data repository. A code may indicate which data repository the object is to be retrieved from especially if there is a plurality of remote repositories available such as if the user is roaming in a different mobile network, abroad, for example. Secondly, there is also the extracted data block **382** for extracted data. Typically, such data extraction may be performed, for example, in the extraction block **306** of some application.

FIG. **4A** shows a simplified structure of the MD DB server **240**. The MD DB server **240** is the gatekeeper for personal content. This means that it is the element where access restrictions and other confidentiality issues are considered. When requesting a service from the MD DB server, the user can set different access policies for different parts of the data. In addition, access to a specific type of content may be restricted for some service and service application, whereas some other application may access that same data. Also, policies such as read only, read only at the MD DB server, or similar solutions may be implemented. The purpose of the latter example is to allow third parties to supply various analysis and service applications while maintaining the privacy of the user by disallowing the misuse of confidential or strictly personal information. In other words, the MD DB server is partially intended for managing objects and information extracted and/or

generated from said objects, the objects and information being the personal content stored in the remote data repository.

One aspect in the handling of confidentiality issues is discussed below with reference to FIG. **12**. Preferably, the server has a daemon **402** to activate the correct service provision block **412**. For this purpose both the daemon **402** and the service provision block **412** have definitions **404** which contain, for example, information on requirements posed by the services and different options for service requests. In order to complement this task, the MD DB server may also have data extraction means in an extraction block **406**. The extracted information must be associated to the corresponding personal content or content object. Therefore, the system also has association means in the corresponding association block **408**. Due to the possibly large number of personal content items, the system may further comprise selection means in a selection block **410** responsible for the appropriate selection of personal content, either in the form of an object or extracted data, during the provision of the personalized service.

FIG. **4B** is a schematic block diagram of the functional blocks of an exemplary remote data repository **242**. First of all, the repository contains personal content in the form of objects in the object register **452**. The repository may also contain a summary or several summaries **456** of the content stored in the register. Data extracted **454** from the objects and also data generated **458** by some services may be a part of this content. The general term "services" is here to be understood as provisions of means for analysis and the combining of information so that the personal content of the user may be enhanced at least in some manner. It is to be noted that the services may be provided in the service provision block **412** of the MD DB server, in a separate application server **250** or **251**, or both.

The storage means **202**, together with user terminal **100** hardware HW and software SW blocks, have a file system. The file system may be arranged so that the HW/SW block requests an object including stored personal content from the storage means in the mobile terminal. If the requested object is not located in said storage means, the system, in practice the HW/SW block, may be further arranged to request the object from the remote data repository. This may be implemented so that there is a list of locally available objects in the terminal, and a list of remotely available objects, i.e. objects that have already been transferred to remote data repository **242**. If

the object is located in the remote repository, instead of delivering the object the local storage means **202** may return a return code, in a predetermined format. This operation may be a result of a read request for the object.

The production or provision of services may also be performed in steps in such a way that some parts of the service are generated in one server and some other parts in another. It is clear that the design of the services gets a bit more complicated when the number of servers involved increases. Finally, the service may be combined from several parts to form the complete enhanced content, and the combination can be performed either at some server in the system or at the user terminal. In the latter case the combination of the content from parts may be virtual so that the user cannot recognize how the different parts of the content are actually produced.

FIG. **5** is a block diagram of an upload registry **280**. The upload registry is connected to the external world via a communications block **500** which receives messages (**L7**) intended for the upload registry and sends messages to user terminals, for example. The upload registry **280** may include definitions **502**, such as access policies, lists of allowed users and their services, pricing policies and cost structures of the mobile network, and so on. The upload registry may also contain registrations **504**, which are on a per user basis and show the personal content objects registered for uploading, for example. The upload registry may be used for downloading files as well. The monitor block **506** monitors the conditions for each user, the conditions preferably being stored in the definitions. When a predetermined condition is met, such as some threshold value for transfer price per data unit, the transfer may be initiated by informing the notificator **508**. The notificator generates a message **L9** to be sent to the mobile terminal daemon **322** and then passes it to the communications block **500** to be delivered further.

FIG. **6** shows various aspects of the present innovative concept. The presented mechanisms provide significant advantages in view of user interface and ease of use, and also take into account cost efficiency and radio network utilization considerations when providing a mobile user with storing and content enrichment services.

First, the dashed box **61** illustrates some tasks prior to the transfer of the personal content. When the user or the terminal acquires some personal content, it is detected (step **602**) by the hardware **200**, and the hardware notifies (message **L1**) the terminal daemon **326** provided within the

terminal. The terminal daemon is a terminate-and-stay-resident-type application which wakes up when receiving a notification. The terminal daemon analyzes the notification by, for example, checking which kind of content was acquired, and then it decides, partially based on the software capabilities and settings of the terminal, if an application **201** in the terminal is to be woken up by sending message **L3**.

The terminal application **201** is loaded or activated in the terminal. If the application requires a significant computational effort, the terminal may run it with a lower priority or it may wait until the terminal is in an idle state in order to avoid reducing the comfort of use in an annoying way. The application may extract some data from the personal content. For example, if the content in question is a digital image, it may extract (step **604**) parameters such as the time and date of taking the image, exposure and flash settings, and so forth. It may also request some other information related to the content, such as positioning information. If the positioning information describing the user's past behavior is stored in a location history database, the information may be requested from there. Alternatively, the positioning information may be requested from a mobile network positioning system **282**. Also, the data extraction step **604** may include reading values of a register in the terminal the cell identity of the current cell of the terminal, location area information, and so on.

Some other ways to perform the detection and simple data extraction steps may be realized, for example, by implementing the system in such a way that a user indicates his/her wish to use some personalized service at a given moment by simply pressing an activation button in his/her mobile terminal. The pressing of the button may initiate the application responsible for of collecting certain information and, for example, initiate some other applications, such as a digital camera user interface. Thus, when the user presses the button, the terminal system having a digital camera functionality may request the user to take a picture. Then the information relevant to the taking of the picture is extracted.

What kind of information is marked as being related depends on the reliability of the algorithm that identifies two pieces of information as related and on the usefulness of this relationship to the user; the latter is decisive because the storing of each relationship takes up valuable memory space. As all data is personal, one heuristic algorithm may be applied ac-

cording to which all data that originated at the same time is interconnected. The concept of simultaneity may be further narrowed down on the basis of user preferences and system findings. For example, in some cases a time difference of half an hour between the origins of two objects might still be considered simultaneous, whereas in other cases a gap of five minutes might already be too large. This basic approach can be varied, e.g. according to the types of generated data and, of course, according to the connecting concept; for example, instead of time, location can be used. The information about the relationships to other data is considered to be part of the extracted data. This kind of relationship may be regarded as the association of different objects.

The extracted information, including the information on relationships between pieces of data, is preferably stored (message **L5**) in the terminal database **202**. The terminal database may be a register residing on a memory chip, such as the random access memory of the subscriber identity module, or a terminal memory, or a magnetic device such as a hard disk. Further, the terminal application may notify the upload registry **280** of the file transfer system of the operator to indicate that new content has been acquired and that the content is ready for uploading (message **L7**). Together with this notification, there may be an indicator of the current status of the terminal device, such as the available memory, the estimated charge status of the mobile terminal battery, and so on.

The dashed box **62** shows how the actual transfering of personal content is performed. Indeed, the transfering can be done in many other ways as well, but the inventive concept described herewith is believed to offer significant advantages over the prior art data transfer systems, such as a mobile-oriented circuit-switched packet data connection or a normal packet-switched packet data connection well known in the art. The ideas of automating some tasks, delaying the actual transfer until predefined criteria are fulfilled, etc., as well as the mechanism whereby the transfer is initiated in the upload registry are believed to be inventive.

In step **606** the upload registry **280** monitors the indicators sent by the mobile terminal. It may, for example, take cost efficiency or radio network usage considerations into account. This means that the uploading of the personal content is initiated when the radio network load drops under a predefined threshold, in terms of transfer price per unit of data, relative usage capacity, or available bandwidth. Also, the pricing of the data transfer may be

included in such a way that the transfer is preferably performed only during off-peak traffic hours. However, there may be some specific criteria which trigger immediate transfer, but these considerations are not discussed here.

When the conditions for the uploading are met, the upload registry notifies a terminal daemon **322** by sending a notification message **L9**. The terminal daemon **322** is a functional unit separate from the terminal daemon **326**, in the sense that the terminal daemon **326** is invocable from the applications in the mobile terminal whereas the terminal daemon **322** accepts external notifications. This is mainly because of security considerations, because whereas the part of the application invocable by the former daemon **326** has access to practically all information available in the terminal, the part of the application invocable by the latter daemon **322** has access to only part of the files in the terminal data storage **202**.

After receiving notification **L9**, the daemon wakes up the terminal application **201** defined in the daemon settings (message **L11**). This application may be different from the application **201** referred to earlier, but it can also be implemented using modular programming techniques to restrict the access to information from the corresponding parts as well. The terminal application **201** requests (message **L13**) data from the terminal data storage **202**, for example, reads the terminal memory, and receives the object in a message **L15** including personal content and data extracted from it.

After retrieving the object and data, the terminal application establishes (message **L17**) a connection to server daemon **402** in order to upload the object and data (message **L19**). The server daemon stores the uploaded content by sending it (message **L21**) to the MD DB server **240**, which further stores it in the remote data repository. According to one aspect of the implementation, it is preferable that there is feedback from **240** to **201** and/or **202** showing that the object has been correctly stored, so that objects not correctly stored are not accidentally erased.

The dashed box **63** shows an example of tasks possible after transfering the content and storing it in the remote data repository. The server daemon wakes up different applications if necessary. For example, an analysis application running in the application server **251** may be called by sending a wake-up call **L23**, and a content combination application may be invoked by sending another wake-up call **L25**. In order to facilitate this, the MD DB server **240** may inform the server daemon **402** of the services sub-

scribed to after they have been requested. This way the server daemon may send the wake-up requests **L23** and **L25** directly to the applications, and it is not necessary for the MD DB server to perform this task.

There may also be a plurality of applications. In the dashed boxes **64A** and **64B**, two different kinds of applications are presented schematically. The real applications are basically applications having characteristics in common with either one or both of these two types.

The dashed box **64A** shows some exemplary tasks possibly required for enabling the operation of the application server **251**. As already noticed with reference to the dashed box **63**, the message **L23** which acted as a request for server application **251** was received at the server application **251**. In this case the message **L23** included either the identity of the user requesting the service or the identity of the object to be used for the service, which in this case is generating new data.

The object is fetched from the MD DB server by sending a message **L27**. The object could be fetched from the MD DB as well, if desired, but this example is solely to be understood as an enabling example and not as restrictive in any sense. After the object has been retrieved, it is analyzed in step **616**, and at least partially in response to said analysis step, new data is generated in step **618**. The new data is then stored by sending it in a message **L33**. In the MD DB server there may also be an incremental summary. This must be updated, i.e. the operations performed on the data, and possibly also some results of the analysis can be described. The summary is stored by sending an update request **L37** to the MD DB server **240**.

In the dashed box **64B**, another service application **250** performs similar tasks. This application has been woken up by a message **L25**. It retrieves an object and data by requesting (message **L29**) them from the MD DB server **240**. Then it fetches (message **L31**) external data from an external database **260**. It is to be noted that the retrieved data **L29** does not necessarily need to be analyzed any more because the application may have got the information as to what it is about in the original notification **L25** from the server daemon **402**.

The dashed box **65** which is explained below in more detail with reference to FIG. **15**, performs the (temporary) deletion of obsolete files that have already been transferred to the remote repository. The terminal application **201** sends a request **L51** to the terminal database **202** inquiring the

status of the local data storage capacity. The terminal database sends a storage response **L53** to the terminal application, which in step **651** analyzes the response according to the definitions. If some predetermined criteria are met, a selection step **653** follows, where items selected for deletion are identified, and this is further communicated to terminal database **202** by sending a delete command **L55**.

FIG. **7** shows the action logic of step **602** in cases where the functionality is implemented in the terminal hardware. The hardware performs (step **702**) its other functions, after which the processing is interrupted in order to check (step **704**) whether a new object has been acquired. If the result is that some new object has been acquired, the terminal daemon **326** is notified in step **706**. After this the terminal hardware continues its normal operation. The checking step may be implemented, for example, by reserving some interruption of the mobile terminal CPU for a program performing the checking. Alternatively, a step **706** may be implemented in the storage means **202** of the mobile terminal in such a way that when the object register **380** receives a new item it performs the step **706** simultaneously. This can also be performed in the database accession block **362** or in the object generation block **366**. The daemon may be notified when the block **362** accesses the database in the "create new object" mode, or when the object generation block is generating a new object.

FIG. **8** is a flow diagram of the terminal daemon **326**. When the terminal daemon receives (step **802**) a notification, it wakes up, i.e. it goes into an active state. Basically, this means that the priority or the given processor time of the application is increased, and/or the necessary program code is loaded into the memory from the storage means **202**.

The first thing after receiving the notification is to identify (step **804**) the object. For this purpose, the terminal daemon must either be informed about the kind of object, for example, by providing the wake-up message with the file type identifier or by checking the identifier by the terminal daemon itself. The identifier logic can be similar to that widely used by different computer operating systems (filename extensions or file headers), or the identifier may be selected to correspond, for example, to different applications of a Nokia phone.

When the object has been identified, the next step **806** to be performed is to read definitions of the object type. The daemon may have a list

of applicable analysis means and routines for specific object types. For example, digital images may be of interest to it, while stored short messages are not to be analyzed, and so on. Each object type may have multiple analysis steps to be performed, but this is not necessary. When an analysis application is installed in the terminal device, or when such a service is installed in the MD DB system so that some new sort of content may be analyzed, the terminal daemon is to be informed of this.

If the object is of such a type that there is a need to i) extract data from the object, ii) transfer it to the MD DB for analysis, or iii) simply transfer the object to the MD DB, the corresponding application is woken up (step **810**). After this step the terminal daemon returns (step **812**) to idle status, i.e. starts again to listen for possible notifications.

FIG. **9** is a flow diagram describing the actions of a terminal application **201**. First, the application wakes up (step **902**). This is preferably accompanied with reading definitions in step **904**. The definitions may include preferences for the analysis task, for example, when a digital image is in question, i) whether optical character recognition is to be performed, ii) what data is going to be extracted, and iii) whether the position of the mobile station is to be found out with respect to the analysis. This kind of setting information may be incorporated in the definitions table or file. After reading the definitions, the object is read in step **906** into the terminal memory.

Data extraction is performed (step **908**) for the object in accordance with the definitions. As already stated, this includes the detection of relationships with other personal objects. In the next step **910** the extracted data is stored in the extracted data block **382** of the storage means **202**. Then, in step **912** it is checked whether or not more extraction analysis is to be performed on the object according to the definitions. In a positive case the control returns to step **908**; in the opposite case the object and the extracted data are registered (step **912**). Then the execution of the terminal application is ready for completion (step **916**).

The registration step **914** includes notifying the upload registry **280** about the content acquired. This may be performed, for example, by sending a short message, a data packet, or some other suitable information carrier to the upload registry.

FIG. **10** is a flow diagram of the actions of the upload registry **280**. In accordance with some aspects of the solution, the upload registry is favorably dedicated for several mobile terminals. When the upload registry

favorably dedicated for several mobile terminals. When the upload registry application is initiated, the application reads the definitions in step **1002**. Then the application starts listening and waits (step **1004**) for interesting events. For example, a message **L7** may be received. Basically, the upload registry has some conditions whose fulfillment it is monitoring. If there are already some registered objects, the listening may involve checking the system time, checking the network traffic pricing parameters, etc. Or, alternatively, the application may just wait a predetermined period of time.

In step **1006** it is checked whether the predefined criteria are met. The criteria may include i) the transfer price, ii) the radio network utilization coefficient, iii) the location of the user, i.e. the data is transferred only if the user is roaming in the home PLMN of the user, iv) forced transfer, in which case the data is transferred without taking the other criteria into account, and v) any combination of the above.

If some criterion is met, the upload registry notifies the network reachable daemon **322** of the mobile terminal. After this step the upload registry is ready to serve the next client, i.e. return to step **1002**. In some respects, it is very important that the registry is capable of serving virtually several (even thousands) of clients at the same time. The upload registry may be implemented, for example, using a computer running UNIX-like operation system, whereas the registry functionality may be implemented by CRON-like program in combination with some network reachable daemon that writes the registrations into a file, the system periodically checking the registrations and then performing operations if necessary. A registry implemented using a simple Intel Pentium III processor family (both are registered trademarks by Intel), for example, may easily handle thousands of users. This is usually not only a question of the processor, but also of the internal and external bandwidth.

If no criterion is met to initiate the transfer, then a check is made (step **1008**) as to whether a registration has been requested. If no registration has been requested, the control is returned to step **1002**. In the opposite case, the registration is read (step **1010**), and stored (step **1012**). The registration may be further analyzed if the definitions need to be updated. The definitions are updated (step **1014**) when necessary. After this step the control is returned to the step **1002**.

FIG. **11** illustrates the terminal daemon **322**. The daemon receives a notification in step **1102**, preferably from the upload registry and not from a hacker, checks the notification, and if it is in a predetermined form, i.e. if the optional password and originating number or source address are correct, it accepts it and then wakes up (step **1104**) the corresponding transfer application **201**. After this the terminal daemon returns to an idle status (step **1106**), i.e. starts to wait for the next notification.

FIG. **12** is a diagram showing an example of a control flow of the terminal application **201**. The terminal application is woken up in step **1202** after a notification message **L11** from the terminal daemon **322** is received. The application reads the definitions (step **1204**), possibly including any in the wake-up message which the daemon has passed to the application and any which may be originating from the upload registry and specifying, for example, i) the transfer mode, ii) the destination address, and iii) the possible tasks before transfering the data.

The terminal application **201** establishes (step **1206**) a connection to the server daemon **402**, which preferably is located in the MD DB server **240**. After this the terminal application transfers to the MD DB daemon objects and extracted data (steps **1208** and **1210**, respectively) . As soon as the transfer of the objects and extracted data has been completed, the application checks (step **1212**) whether there is still something to be transferred, and if necessary returns to steps **1208** and **1210**. This may be the case, for example, when the user is simultaneously using the terminal equipment for acquiring new personal content. When necessary the content recently acquired may be transferred as well.. Finally, the terminal receives some feedback as to which data has been safely transferred into the MD DB server **240**. It is also possible to include transactional mechanisms here that ensure an "all-or-nothing" behavior: either all objects are stored in the server DB or, if the storage of at least one object fails, no object that is part of the same transaction is stored.

After the transfer has been completed, the connection is closed (step **1214**) and the object register **380** and extracted data **382** may be updated (step **1216**) to indicate the contents that have been transferred. After this, the application **201** is ready with its tasks or execution is terminated (step **1218**).

FIG. **13** is a flow chart illustrating an example of the operation of the server daemon **402**. Firstly, the daemon receives (step **1302**) a connection request from the MD application **334**. It reads (step **1304**) definitions associated with the current client and then opens the connection to the terminal application in step **1306**. Preferably this is performed by accepting the connection request sent by the MD application, but the connection may be opened from the server as well. The objects and data are transferred in step **1308** in a manner corresponding to the scheme presented in FIG. **12**. If the transfer has not been completed (as checked in step **1310**) the connection is not closed, but further transfer is commenced in step **1308** until everything has been transferred. Then a receipt concerning all correctly transferred objects is issued to the terminal application, the connection is closed (step **1312**), and the daemon opens (step **1314**) another connection to the MD DB server **240** or the remote data repository **242**, depending on the implementation. The objects and data are transferred (step **1316**), and similarly, if there is something else to be transferred (this is checked in step **1318**), the transfer step **1316** is repeated until everything has been transferred prior to closing the connection **1320**; if the transactional mode has been activated, all changes are revoked as soon as one object cannot be stored. Then the service applications are notified in step **1322** before the server daemon goes (step **1324**) into a sleep state. If the terminal application has requested this behavior, a message is sent to it that contains information about the identity and/or number of correctly stored objects; this behavior may also be implemented as part of one of the aforementioned service applications.

FIG. **14** presents the operation of the application in the MD DB server, or alternatively of a service application. The application is woken up in step **1402** after receiving a notification (message **L21**) from the server daemon **402**. The application reads (step **1404**) related definitions **404** from the definitions file and a summary **456** from the remote data repository.

After performing the previous steps, the application transfers (step **1408**) objects from the object register **452** and the extracted data from extracted data part **458**. One possibility for the analysis step **616** is that the transferred objects are read (step **1410**) and data is extracted (step **1412**) from said objects. The extraction of data means here extracting date and time information and/or other similar information, such as exposure parameters if the object is a digital image. If the object is a short message, a multi-

media message, or something similar, the extracted information may also include information about the sender, such as the MSISDN or phonebook entry information. If the object is a video or audio clip, the information may include some other parameters, such as the duration of the clip, the bit rate, the copyright owner of the corresponding clip, etc.

Then it is decided (step **1414**) whether or not external data is needed. If the purpose of the application is such that no external data is needed, but only new data is generated, the processing continues in step **618**. In the opposite case, data is retrieved (step **1416**) from an external database. Step **1416** may in practice mean several iterations of this step, so that also step **1414** is executed to check whether even more external data is considered necessary. The parameters for the fetching procedure may be selected from the extracted data, and if some privacy control mechanism is needed, the parameters can be checked in the MD DB system as to whether they contain any information interfering with user privacy. Basically, if the object is a digital image and the extracted information includes information such as the date and time of the image, and positioning information relating to the geographical location where the image was taken, this data can be used as parameters to search information from an Internet search engine, for example. The data may thus be used as parameters to a query, or in some other similar manner.

In step **618** new information is generated. This may include, for example, performing optical character or text recognition, and voice-to-text conversion. This information may again lead to the determination of new relationships between objects in the user's personal data. For example, if the object under consideration is a GPS measurement, a possible server application could find the street address of this GPS location; if this street address matches one of the addresses in the user's database of contacts, a match can be made between a contact entry and a GPS measurement and any image that the user has taken at this location, for example. Step **618** may also include performing some steps to merge the retrieved information and the already extracted data with the generated data. In some cases, this might mean the simple enlargement of available metadata, whereas in other cases some data might be discarded when there is more precise information available; e.g., the information "Southern Finland" might be discarded for an

image if further analysis shows that it was taken at the "Olympic Stadium" in Helsinki.

When the new data has been generated, it is ready to be stored. This is performed in step **1420** and includes storing the data in the generated data part **458**. Then the summary **456** has to be updated. This procedure is performed preferably in the application, which has read the summary in step **1406**. The summary is updated to show the processes performed on the objects, to show the association between the objects and the generated and extracted data, and so on. The updated summary is stored (step **1424**) in the summary **456** located in the data repository **242**. Then the execution of the application is ended (step **1426**).

FIG. **15** shows an example flowchart of how the deletion is performed in the mobile terminal. Preferably the deletion part is located in the MD application **334**. First in step **1502** the definitions regarding the deletion of objects are read. The definitions may include classification of candidate object types to be deleted, some additional predetermined criteria, such as the required minimum age of the objects to be deleted, etc. The definitions may also include some limits for terminal memory storage (i.e. free space in storage means **202**), such as upper and lower storage limits. Further, the definitions may include conditions related to these limits. The upper storage limit is the critical upper threshold for the terminal database capacity, the point at which the terminal is very soon going to run short of storage capacity. Then the forced transfer of some objects would be advantageous so that they can be locally deleted to create space in the terminal memory. The lower storage limit is the point at which it might already be preferable to delete at least some objects, but the situation is not considered a particularly critical one so that no forced transfer is performed.

The terminal application waits (step **1504**), for example, a predetermined time defined in the definitions, prior to commencing any tasks. After the predefined period has elapsed, the application may read (step **1506**) the object register and the analyzed data status. The application may request (step **1510**) the storage status from the terminal database **202** (message **L51**). In step **1512** the application receives a storage response (message **L53**). The response is then analyzed, i.e. step **651** is performed.

If some objects have already been transferred (which is checked in step **1508**), the application compares (step **1514**) the storage response with

the lower storage limit. If the limit has been reached, the objects to be deleted are selected (step **1516**) and then deleted (step **653**). In more detail, if the limit has been reached, possible candidates for deletion are determined based on upload status and access frequencies; if no such candidates, or not a sufficient number, can be determined, the operation skips forward back to step **1502**. Otherwise it proceeds to step **653**.

If the limit has not been reached, the operation returns to step **1502**. The idea behind performing steps like this is that there is no need to delete the objects which are transferred before some part, say 30%, of the storage capacity of the mobile terminal has been used.

If it turns out in step **1508** that no objects have been transferred yet, the application compares (step **1532**) the storage response with the upper storage limit. Here if it turns out that the upper limit has not been reached, the application returns to step **1502** and commences to wait after reading the definitions in order to detect possible updates or modifications. In the opposite case, however, a transfer is forced (step **1534**), which may include waking up the transfer application **201** in the terminal by sending an emergency code which initiates an immediate transfer as described in FIG. **12** onward from step **1202**. After the transfer has been completed, i.e. after step **1218**, the execution returns to steps **1516** and **653**, where the objects to be deleted are first selected and then deleted. In step **653** a message **L55** is generated, which is then fed into the terminal database so that the files in the terminal database are erased. The actual implementation of this depends on the system used, but basically the principle that the files to be deleted are first selected and then deleted is applicable in general.

It is possible that the user modifies some attributes of the files so that the deletion of the items in question may be prohibited. Also, the user may have an option to confirm automatic deletion proposed by the application.

The enhanced storing of personal content can be integrated into the system in many ways. Generally, a remote repository belonging to the Media Diary framework can be any accessible database. The server-client system described may be implemented in various ways as well. The applications in the mobile terminal, the upload registry, and the remote repository may be implemented in different functional units as well, even in such a way that the order of the steps presented in the preferred embodiment differs. This does

not change the idea of the present invention which is also applicable in such cases. For example, with regard to setting different definitions and defining the user interface, the service applications can be utilized either by using a service programming interface with any compatible programming language or by using any service user interface described by a generalized description language. Different kinds of adapters can be implemented for service integration.

One neat example of the invention reduced into practice is that the short messages, multimedia messages, or emails are transferred to the remote data repository. Some data, such as originator, recipient and subject of the message may be extracted, and even some other data, such as location information of the user may be stored. In this way the user might get some benefit of the idea that a specific message was received in a specific point in time, such as when the user was travelling by train from Cologne to Munich, Germany. This kind of metadata may enrich the personal content. Further, the data can be searched on the basis of the metadata. This approach also clearly helps the user to reduce the drawbacks of the finite terminal memory, for example.

Although the invention was described above with reference to the examples shown in the appended drawings, it is to be understood that the invention is not limited to these, but that it may be modified by those skilled in the art without departing from the scope and spirit of the invention.

Claims

1. A system for providing access to stored objects for mobile users, the system **comprising**

- a mobile terminal provided with means for acquiring personal content, the mobile terminal being adapted to be in wireless communication with a telecommunications network,

- storage means in the mobile terminal, the storage means being adapted to store the personal content acquired,

- at least one remote data repository connected to the telecommunications system for storing personal content, whereby at least one of the repositories is assigned for the use of each mobile terminal,

- means adapted to transfer selected personal content between the remote data repository and the storage means through said telecommunications system, the means to include predetermined criteria, the fulfillment of which initiates said transfer,

- the mobile terminal further provided with means for accessing stored personal content wherein i) the means are adapted to request an object including stored personal content from the storage means in the mobile terminal, ii) the storage means are adapted to respond with a predetermined return code if the requested object is not located in said storage means, and iii) the means are further arranged to request the object from the remote data repository if the return code indicates that the requested object is not located in said storage means.

2. A system according to claim 1, the system further **comprising** a server connected to said remote data repository, for managing said objects and information extracted and/or generated from said objects, the objects and information being the personal content stored in the remote data repository.

3. A system according to claim 2, the system further **comprising**

- means in the server for updating the information related to said object to indicate that the object has been requested by the mobile terminal and

- means for storing the updated information in the remote data repository.

4. A system according to claim 1, the system further **comprising** means in the mobile terminal for updating a register of the objects and/or extracted data stored at least at one point in time in the mobile terminal storage means.

5. A system according to claim 4, **wherein** said updating includes marking deleted and/or transferred objects and/or extracted data that has been transferred to the remote data repository.

6. A method for providing access to stored objects for mobile users, the method **comprising**

- acquiring personal content in the mobile terminal adapted to be in wireless communication with a telecommunications network,

- storing the personal content acquired in the mobile terminal,

- assigning at least one remote data repository for the use of each mobile terminal, the repository being connected to the telecommunications system for storing personal content,

- transfering selected personal content between the remote data repository and the mobile terminal through said telecommunications system, the means to include predetermined criteria, the fulfillment of which initiates said transfer,

- accessing stored personal content from the mobile terminal by i) requesting an object including stored personal content, ii) receiving a predetermined return code if the requested object is not located in the mobile terminal, and iii) further requesting the object from the remote data repository if the return code indicates that the requested object is not located in the mobile terminal.

7. A method according to claim 6, the method further **comprising** the step of connecting a server to said remote data repository for managing objects and information extracted and/or generated from said objects, the objects and information forming the personal content stored in the remote data repository.

8. A method according to claim 7, the method further **comprising** the steps of

- updating the information related to said object to indicate that the object has been requested by the mobile terminal and

- storing the updated information in the remote data repository.

9. A method according to claim 6, the method further **comprising** the step of subsequently updating a register of the objects and/or extracted data stored in the mobile terminal storage means.

10. A method according to claim 9, **wherein** said updating includes marking deleted and/or transferred objects and/or extracted data that has been transferred to the remote data repository.

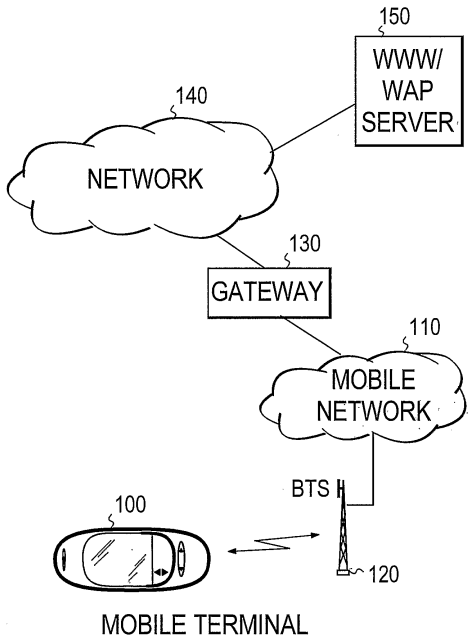


FIG. 1 (PRIOR ART)

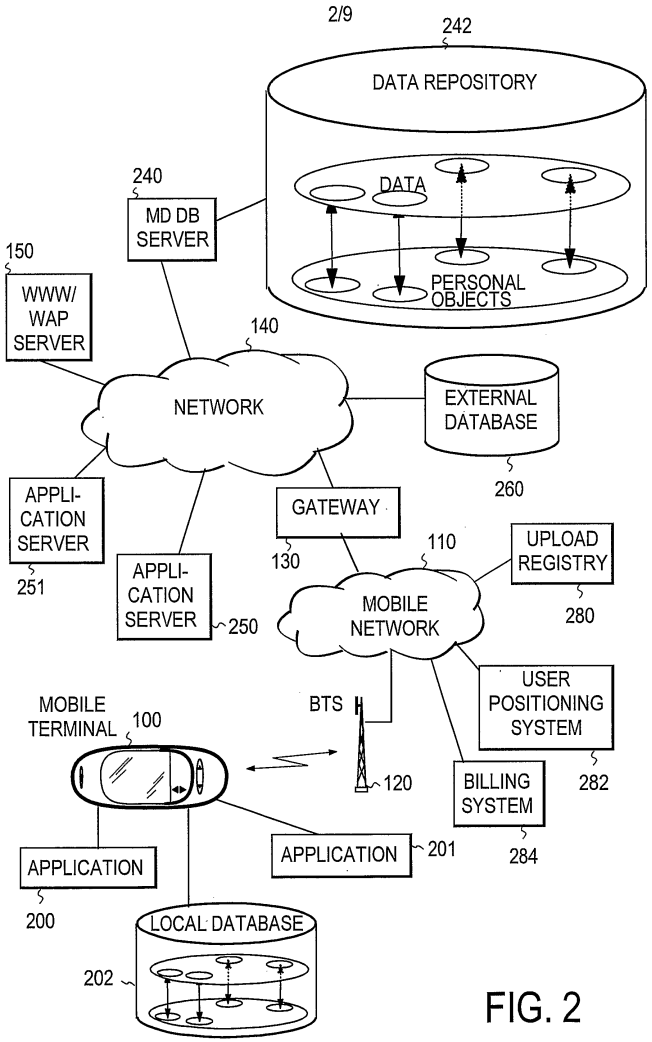


FIG. 2

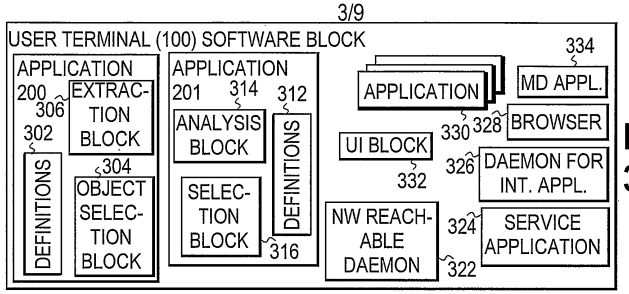


FIG. 3A

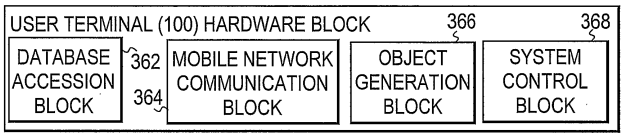


FIG. 3B

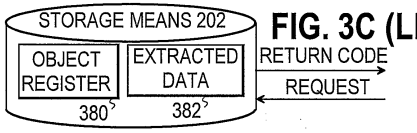
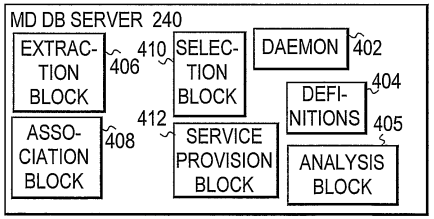


FIG. 3C (LEFT)

FIG. 4A (BELOW)



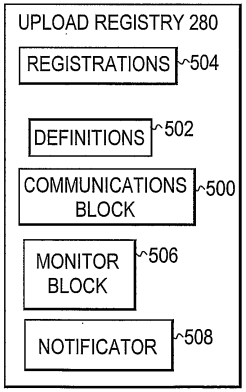


FIG. 5 (ABOVE)

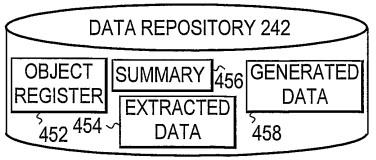


FIG. 4B (LEFT)

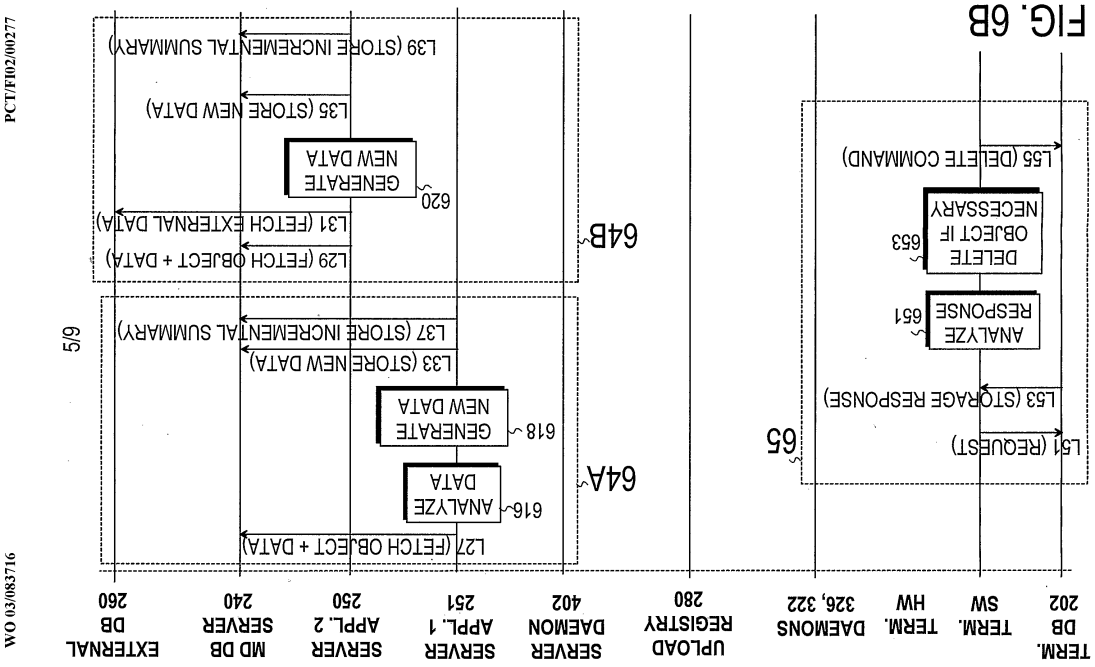


FIG. 6B
TERM. DB 202
TERM. SW
TERM. HW
DAEMONS 326, 322
UPLOAD REGISTRY 280
SERVER DAEMON 402
SERVER APPL. 1 251
SERVER APPL. 2 250
MD DB SERVER 240
EXTERNAL DB 260
L61 (REQUEST)
L53 (STORAGE RESPONSE)
ANALYZE RESPONSE
651
DELETE OBJECT IF NECESSARY
653
L55 (DELETE COMMAND)
65
64A
64B
5/9
L27 (FETCH OBJECT + DATA)
616
ANALYZE DATA
618
GENERATE NEW DATA
L33 (STORE NEW DATA)
L37 (STORE INCREMENTAL SUMMARY)
L29 (FETCH OBJECT + DATA)
L31 (FETCH EXTERNAL DATA)
620
GENERATE NEW DATA
L35 (STORE NEW DATA)
L39 (STORE INCREMENTAL SUMMARY)
PCT/FI02/00277
WO 03/083716

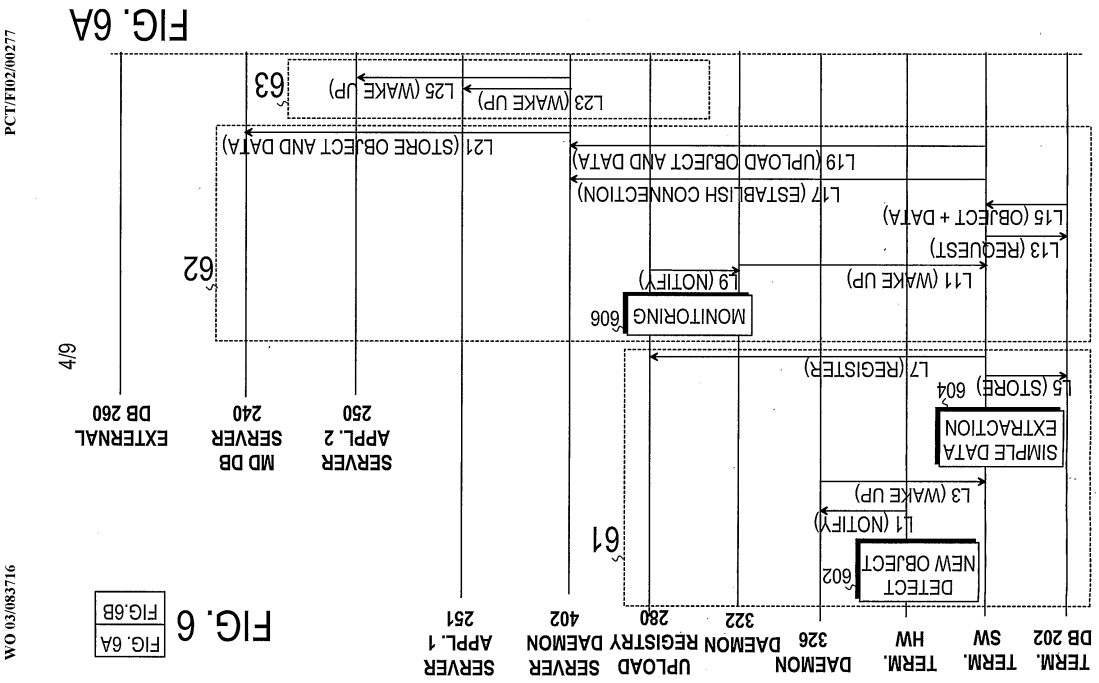


FIG. 6A
TERM. DB 202
TERM. SW
TERM. HW
DAEMON 326
DAEMON 322
UPLOAD REGISTRY 280
SERVER DAEMON 402
SERVER APPL. 1 251
SERVER APPL. 2 250
MD DB SERVER 240
EXTERNAL DB 260
DETECT NEW OBJECT
602
61
L1 (NOTIFY)
L3 (WAKE UP)
SIMPLE DATA EXTRACTION
L5 (STORE)
604
L7 (REGISTER)
MONITORING
606
L9 (NOTIFY)
L11 (WAKE UP)
62
L13 (REQUEST)
L15 (OBJECT + DATA)
L17 (ESTABLISH CONNECTION)
L19 (UPLOAD OBJECT AND DATA)
L21 (STORE OBJECT AND DATA)
L23 (WAKE UP)
L25 (WAKE UP)
63
4/9
FIG. 6
FIG. 6A
FIG.6B
PCT/FI02/00277
WO 03/083716

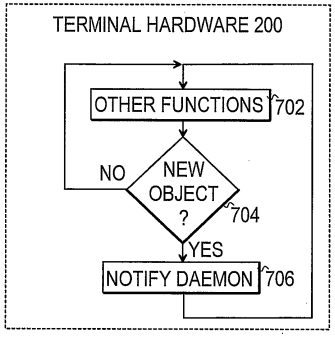


FIG. 7

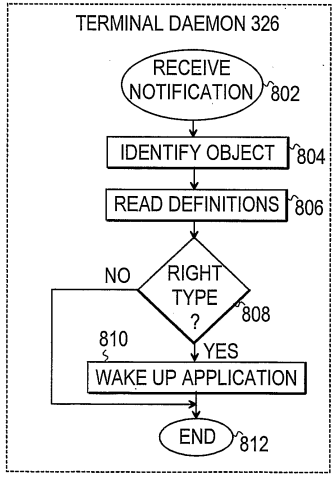


FIG. 8

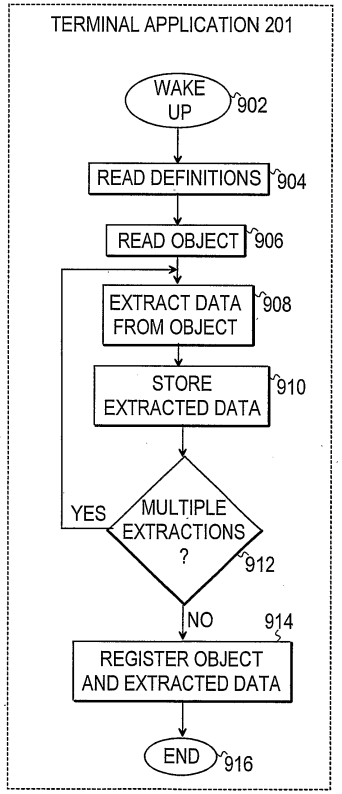


FIG. 9

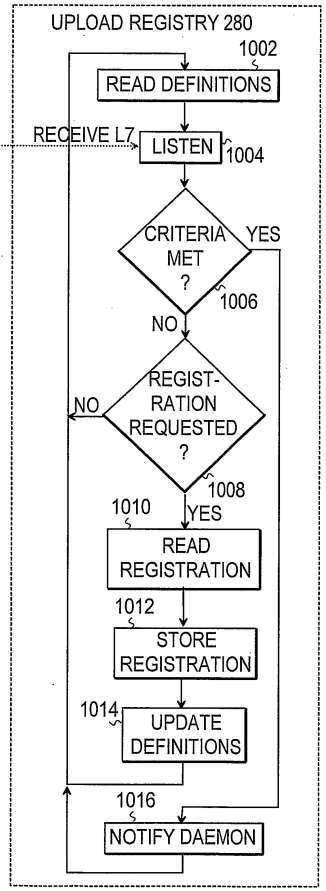


FIG. 10 (ABOVE)

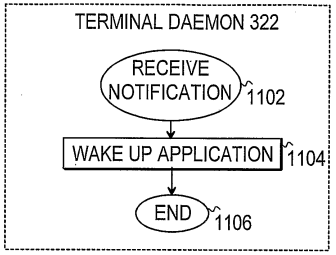


FIG. 11 (ABOVE)

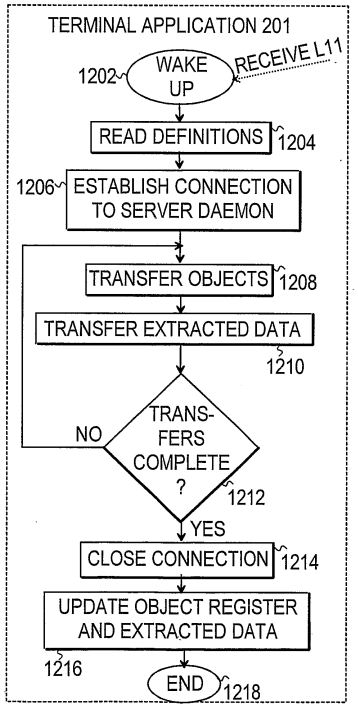


FIG. 12 (RIGHT)

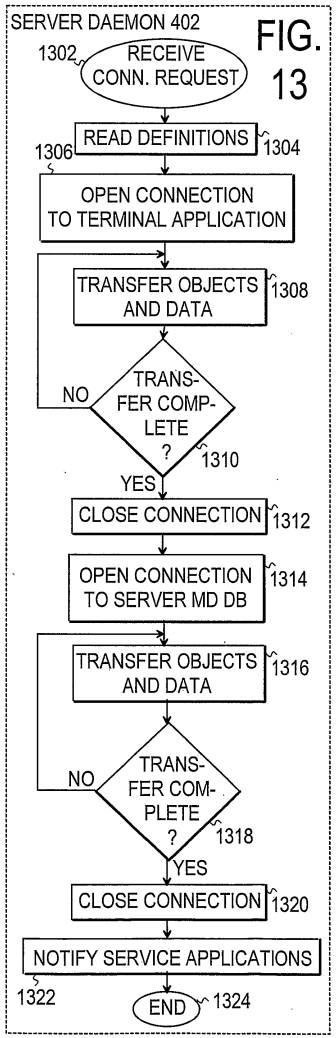
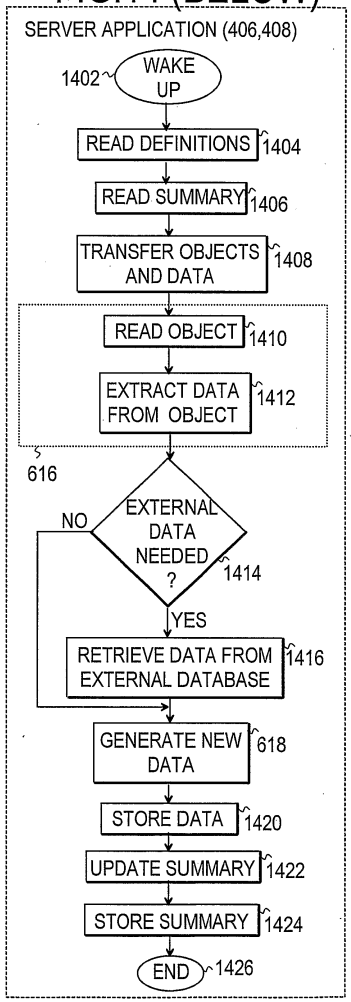


FIG. 13

FIG.14 (BELOW)



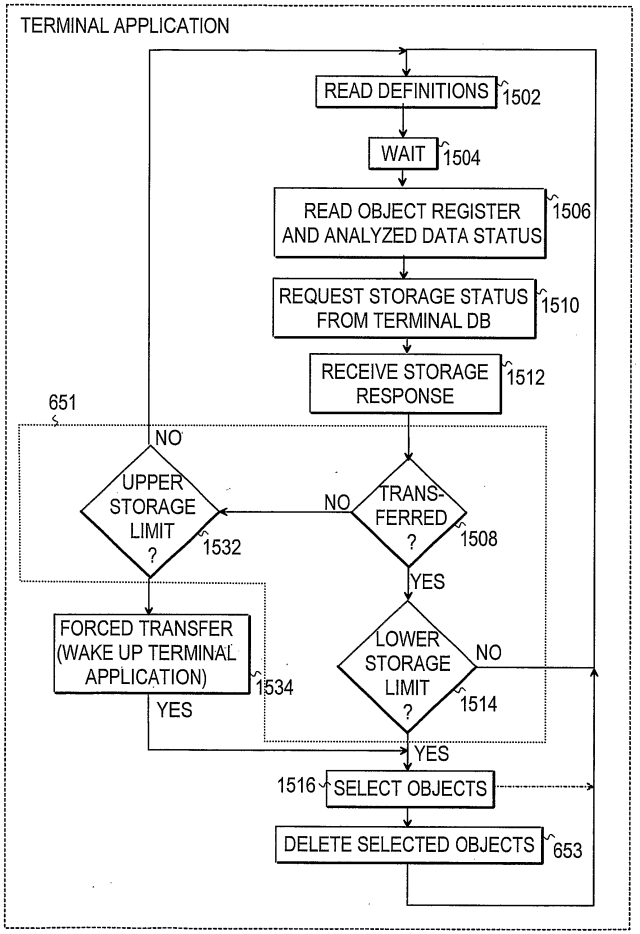


FIG. 15

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 02/00277

A. CLASSIFICATION OF SUBJECT MATTER

IPC7: G06F 17/30
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC7: G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 0063801 A1 (TONI DATA, LLC), 26 October 2000 (26.10.00), page 1, line 12 - line 13; page 15, line 26 - page 16, line 5; page 17, line 17 - page 18, line 14, abstract ---	1-10
A	US 5367698 A (WEBBER, N.F. ET AL.), 22 November 1994 (22.11.94) ---	1-10

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
24 October 2002	05 -11- 2002
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Oskar Pihlgren/LR Telephone No. +46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT
Information on patent family members

30/09/02

International application No.
PCT/FI 02/00277

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 0063801 A1	26/10/00	AU 4641300 A	02/11/00
US 5367698 A	22/11/94	NONE	

Form PCT/ISA/210 (patent family annex) (July 1998)

4

Classification de données multimédia, bases de données et systèmes distribués

Ce chapitre présente un second volet de mes travaux. Il concerne, lui aussi, des questions de classification de données multimédia au moyen de mélanges de lois, mais en lien avec les systèmes de gestion de base de données, ou avec les systèmes distribués. Cette inflexion dans ma trajectoire de recherche, tout en capitalisant sur mes travaux antérieurs, est motivée par la conviction qu'il y a là une branche majeure du "sens de l'histoire" en indexation multimédia. La section 4.1 argumente en ce sens. Ce virage vise, simultanément, à assurer une cohérence des travaux des équipes de recherche auxquelles je participe, Atlas-GRIM (Laboratoire LINA) et Atlas (équipe-projet INRIA), comme décrit en section 4.2. La définition des directions s'est faite aussi par complémentarité vis-à-vis d'autres équipes de recherche, en particulier nationales.

Nous présentons ensuite une synthèse des travaux que nous avons initiés sur ces questions, concernant respectivement :

- l'estimation distribuée de mélange de lois (section 4.3, correspondant au travail de thèse d'A.Nikseresht)
- la construction d'index pour accélérer la recherche parmi un grand nombre de modèles probabilistes de type "mélange de lois" (section 4.4, correspondant à ma contribution au co-encadrement à la thèse de J.Rougui)

Pour chaque cas, un article représentatif est joint en fin de chapitre.

4.1 Evolutions en indexation multimédia par le contenu

La décennie 1990-2000 a vu de nombreux travaux en reconnaissance des formes appliquée à l'image et à la video : détection et reconnaissance de visage, de gestes, de texte,.... La communauté s'est alors beaucoup intéressé à la caractérisation et la discrimination d'*une* classe. Les répercussions en indexation par le contenu ont naturellement été importantes, puisqu'il s'agit d'une application assez directe. La décennie actuelle capitalise sur ces connaissances pour remettre en scène un des principaux et anciens rêves de la vision par ordinateur, temporairement mis de côté de par sa difficulté : la capacité à reconnaître des objets (statiques), des vidéos (caractérisées par leur dynamique), de façon générale. Le virage est celui d'un double passage à l'échelle : en termes de variété visuelle des classes (Ponce, Hebert, Schmid & Zisserman 2006), et en termes de volume de données. Deux problèmes doivent être résolus :

- **développer des techniques permettant de reconnaître qu'un objet appartient à une classe**, alors que la variabilité d'apparence est potentiellement grande, à l'intérieur d'une classe mais aussi entre deux classes, ce qui exclus des techniques trop spécifiques à un type d'objet, et alors que les classes peuvent être très nombreuses. Pendant la décennie 1990-2000, divers descripteurs d'image ont été étudiés, et il apparaît que les ensembles de descripteurs locaux sont majoritairement employés, parfois accompagnés d'une topologie caractérisant l'organisation spatiale de ces descripteurs. Des améliorations successives ont permis un meilleur choix du type de zone locale à considérer, un pouvoir discriminant croissant des descripteurs calculés sur ces zones, une meilleure invariance à l'illumination et au changement de point de vue (notamment à l'échelle), une parcimonie croissante de la représentation. Une image est alors caractérisée par un "sacs de mots visuels", par analogie avec les "sacs de mots" employés en recherche d'information textuelle. Pour des applications spécifiques, néanmoins, couleur, texture et forme ont prouvé leur pertinence (Nilsback & Zisserman 2006).
- **alimenter les techniques ci-dessus, qui requièrent un apprentissage au moins partiellement supervisé, en exemples étiquetés**. On souhaite bien entendu automatiser cette étiquetage, même s'il est incomplet ou partiellement erroné. Des solutions à ce problème sont étudiées :
  - **l'apprentissage faiblement supervisé** : il suffit ici de fournir un étiquetage global à l'image, qui est alors indiquée comme contenant une instance de la classe à apprendre, ou ne la contenant pas (Schmid 2004). Le point clé est que l'instance d'intérêt est susceptible d'être présente dans un "clutter" d'autres éléments visuels. A partir d'un ensemble de tels exemples étiquetés, les descripteurs locaux apparaissant communs entre images "positives" peuvent être identifiés, tandis que les autres sont identifiés comme caractéristiques du "clutter" et délaissés. L'effort d'annotation par l'extérieur du système est donc léger et offre des perspectives applicatives réelles d'automatisation :
    - sur la ressource existante que forment les pages web où texte et image sont entrelacés, l'étiquetage des images au moyen du texte environnant peut être réalisé par une analyse statistique conjointe (Ferecatu, Boujemaa & Crucianu 2005, Sclaroff, La Cascia, Sethi & Taycher 1999),
    - l'interaction entre un utilisateur avec de moteur de recherche d'image basé sur

le texte environnant (par ex. Google Image) est riche d'information, notamment le choix des images effectivement visualisées par l'utilisateur, parmi celles que le moteur retourne.
  - la vidéo fournit également un moyen de localiser des élements à apprendre, puis d'en observer la variabilité d'apparence (Ramanan, Forsyth & Barnard 2006).
- **l'apprentissage semi-supervisé** : il s'agit de tirer parti d'un jeu de données dont seule une partie est étiquetée ; contrairement au cas supervisé, les données non étiquetées sont, elles aussi, exploitées pour affecter les caractéristiques des classes (Cozman, Cirelo, Huang, Cohen & Sebe 2004, Chapelle, Schï¿½lkopf & Zien 2006). Il peut venir en complément de l'apprentissage faiblement supervisé, par exemple pour pouvoir prendre en compte des images collectées sur la toile, qui ne disposeraient pas d'environnement textuel.

Le propos ci-dessus cherche à montrer une évolution de l'indexation multimédia vers des "masses de classes" (voir par ex. (Philbin, Chum, Isard, Sivic & Zisserman 2007)). D'ailleurs, alors que la décennie précédente avait largement mis l'accent sur l'extraction d'attributs dans les média, on observe maintenant une sollicitation croissante des techniques d'apprentissage pour les applications multimédia[16], d'une part, et de nombreux indices de rapprochements entre communautés images et bases de données (la première requiert le savoir-faire de la seconde en termes de structures de données pour l'indexation, la seconde est à la recherche de problèmes ouverts par des données aux caractéristiques nouvelles).

4.2 Contexte local

L'équipe de recherche dont je suis membre a une double culture "bases de données" et "logique floue". A mon arrivée, mon bagage était essentiellement en analyse d'image, l'idée étant de combiner des compétences "image" avec des compétences "bases de données".

J'ai infléchi dès 2001 ma trajectoire vers la classification/reconnaissance statistique de formes en général, plutôt que l'analyse d'image, pour deux raisons :

- il m'a alors semblé que, dans la perspective de travaux à l'interface bases de données/données multimédia, les domaines qui devaient travailler réellement conjointement étaient les bases de données et la classification de données multimédia, plutôt que le travail "près des pixels" ;
- indépendamment du contexte local de recherche, le bagage nécessaire et les questions intéressantes pour les questions d'indexation multimédia s'avéraient relever de l'apprentissage/classification plutôt que de l'extraction d'attributs dans les images.

Ensuite, l'équipe de recherche Atlas-Grim a infléchi sa direction de recherche vers les systèmes distribués. Ceci est largement dû à notre implication dans le montage du projet INRIA Atlas, à Nantes (rattaché au centre de recherche INRIA Rennes - Bretagne Atlantique), où la gestion

[16] Il est révélateur que, dans les années 2004-2006, deux projets INRIA traitant de vision par ordinateur, qui avaient pris la recherche de données mutimédia comme champ applicatif majeur, ont introduit le terme "apprentissage" dans leur nom (VISTA et LEAR) ; cette tendance apparaît également dans les réseaux européens Muscle et Pascal.

de données dans les systèmes pair-à-pair et grappes sont un thème de recherche central. Si les infrastructures distribuées sont un objet d'étude en soi, elles semblent ouvrir aussi des perspectives importantes pour le traitement du problème du passage à l'échelle en classification et recherche de données multimédia par le contenu. Plusieurs points de vue sont possibles et considérés dans notre équipe :

- **répartir les données judicieusement sur les noeuds** d'un cluster, pour augmenter l'efficacité de recherche par le contenu d'images en maximisant le parallélisme (travaux de J.Martinez (Manjarrez, Martinez & Valduriez 2007)) ; placer des données image sur des noeuds d'un réseau pair-à-pair, selon des critères de similarité entre images, de manière à assurer l'efficacité d'un système de navigation dans une base d'image (travaux de J.Cohen) ;
- **supposer que les données sont initialement distribuées**, de par leur lieu de création. On se situe dans l'optique d'un système distribué à grande échelle, où les sources de données sont supposées autonomes et volatiles. Il s'agit alors de proposer des techniques permettant la collaboration entre représentations de données volumineuses et/ou complexes. Nos premiers travaux s'appuient sur des sous-ensembles flous (travaux de G.Raschia (Hayek, Raschia, Valduriez & Mouaddib 2007)) ou sur des modèles probabilistes (travail de thèse d'A.Nikseresht que j'encadre, exposé plus précisemment ci-dessous). Un point clé est que les échanges entre noeuds n'utilisent que ces représentations parcimonieuses, plutôt que les données elles-mêmes.

En résumé, nous souhaitons nous inscrire dans ce double contexte global/local : d'une part, nombre de problèmes intéressants en indexation multimédia par le contenu sont liés au passage à l'échelle en matière d'apprentissage de caractéristiques de classes ; d'autre part, le contexte d'équipe s'oriente vers les systèmes répartis, qui offre une infrastructure intéressante pour aborder le premier problème.

4.3 Estimation distribuée de densité de probabilité

Nous présentons ici un travail mené dans le cadre de la thèse, en cours, d'Afshin Nikseresht, qui s'inscrit dans la dynamique décrite ci-dessus. Il devrait se poursuivre par la thèse de Fati Berrada, qui débute à l'automne 2007, et il s'inscrit dans le projet régional Miles (2007-2010, voir partie "CV" du document).

La perspective visée est l'estimation collective, à partir de sources distribuées, de la densité de probabilité conditionnelle d'une classe, dans un espace d'attributs numériques continus. Ici encore, nous avons choisi de traiter le cas du mélange de lois gaussiennes : pour leur polyvalence d'utilisation, que nous avons évoquée au début de ce document, mais aussi parce que le cas ouvre des possibilités de résultats intéressants en restreignant les échanges entre noeuds aux seuls paramètres des mélanges.

De plus, nous choisissons de considérer un système où l'information se propage au moyen de rumeur (traduction de 'gossip' ; on parle aussi de protocoles épidémiques (Eugster, Guerraoui, Kermarrec & Massoulïï¿$\frac{1}{2}$ 2003)). Leur caractère très décentralisé leur confère de bonnes propriétés en matière de propagation de l'information à large échelle, proches de celles qu'on voit habituellement attribuées aux systèmes pair-à-pair en matière de stockage : haute dis-

ponibilité, forte résistance aux défaillances et absence de noeud critique, passage à l'échelle en nombre de noeuds (Milojicic, Kalogeraki, Lukose, Nagaraja, Pruyne, Richard, Rollins & Xu 2002, Akbarinia, Martins, Pacitti & Valduriez 2006)).

La propagation par rumeur permet de diffuser, mais aussi d'agréger de l'information. Le corpus de connaissances en algorithmique et calcul distribués forme clairement là un fondement (Santoro 2006). Des garanties probabilistes (rapidité, précision) sur des agrégations de statistiques par protocoles épidémiques ont été établies (Boyd, Ghosh, Prabhakar & Shah 2006, Kempe, Dobra & Gehrke 2003). Proches de notre objectif, des versions distribuées de l'algorithme EM ont déjà été proposées (Kowalczyk & Vlassis 2005, Nowak 2003), mais elles travaillent au niveau des données (gossip sur l'étape M), alors que nous souhaitons n'échanger que des paramètres. Evoquons enfin les réseaux de capteurs qui, si leur champ applicatif diffère ce que nous discutons ici, traitent aussi des questions d'estimation, d'apprentissage et de classification sur des données distribuées (Nowak 2003, Donati & Le Cadre 2006).

Réaliser l'estimation d'un mélange de lois collectivement peut impliquer, entre autres, une fusion de représentations issues de différents noeuds et un classement distribué des "performances" des différents noeuds. Nous nous en tiendrons ici à la fusion de représentations. Indépendamment du scénario "distribué", combiner des modèles en vue d'améliorer la performance de classification n'est pas un problème nouveau. Cette combinaison peut se faire de façon paresseuse, c.a.d. qu'on reporte au moment de la requête la combinaison des opinions des différents modèles (souvent le cas pour les *comités d'experts*), ou de façon anticipée. Nous privilégions la combinaison anticipée. La combinaison peut aussi soit être réalisée entre paramètres des modèles, soit nécessiter un retour vers les données (*Adaboost*). Notre approche a, pour l'instant, cherché à ne recourir qu'aux paramètres. Enfin, un cadre que nous n'avons pas exploré est celui des *mélanges d'experts*, où la pondération entre les classifieurs (ou densités estimées) est une fonction variant dans l'espace d'attributs. Ceci justifie bien le nom d' "expert", chaque classifieur devenant spécialisé dans une partie de l'espace d'attribut.

Pratiquement, nous nous sommes concentrés ici sur l'agrégation de mélanges gaussiens. On s'en tiendra ici à deux noeuds, même si le propos vaut plus généralement. Soit un mélange formé d'une combinaison linéaire des mélanges entrants (appelons le *mélange redondant*). Notre principal intérêt, pour l'instant, a été dans la réduction de cette combinaison linéaire des mélanges, vers un mélange comportant moins de composantes (ci-après, *mélange réduit*) que la somme des deux mélanges combinés [17]. En effet, ceux-ci décrivant les mêmes sources, le mélange redondant décrit généralement bien la distribution des attributs, mais on peut supposer une certaine redondance entre composantes, et souhaiter en réduire le nombre. Cette réduction de la redondance est importante, car ces opérations d'agrégation vont se succéder, sur le réseau, dans un processus épidémique, et il importe de ne pas faire croître excessivement le nombre de composantes, au détriment du coût de calcul.

Dans l'optique de la classification de nouvelles données au moyen de la densité conditionnelle de classe qu'on tente de construire ici, une quantité déterminante est la vraisemblance calculée pour ces données. Supposons que le mélange redondant modélise parfaitement la

[17] Cette section donne les grandes lignes de l'article suivant, inséré à la fin de ce chapitre : A.Nikseresht, M. Gelgon, *Gossip-based computation of a Gaussian mixture model for distributed multimedia indexing*, accepté pour publication dans IEEE Transactions on Multimedia.

distribution dont sont tirées les données. Le critère pour que le mélange réduit préserve au mieux cette vraisemblance est de minimiser la divergence de Kullback-Leibler entre mélange redondant et mélange réduit. Il n'existe pas d'expression analytique de cette divergence, dans le cas où on compare les mélanges de gaussiennes. Cependant, cette expression analytique existe entre gaussiennes. On peut alors construire une approximation de la divergence de Kullback entre mélanges, en ne tenant compte que des composantes gaussiennes les plus proches. La partie expérimentale du travail de thèse d'A. Nikseresht a montré que cette approximation donnait de bons résultats sur une application de reconnaissance du locuteur, en comparaison d'un calcul plus précis (et plus coûteux) par un procédé de type Monte-Carlo. Avec cette approximation de la divergence, il est possible d'exploiter un algorithme itératif pour optimiser la divergence entre les mélanges. Cette algorithme est en fait, analogue à un algorithme d'optimisation locale de type CM, où les entités à grouper ici les composantes gaussiennes du mélange redondant, représentées par leurs moments.

La divergence obtenue à convergence croît bien avec le nombre de composantes dans le mélange réduit, mais nous avons noté expérimentalement que les pénalisations de type BIC ou AIC s'appliquent ici avec succès pour trouver un nombre de composants correct, c.a.d. généralement semblable à ce qu'on aurait obtenu par estimation directe sur l'union des sources de données.

Nous appliquons alors une propagation par rumeur dans un réseau de noeuds, consistant en une suite d'agrégations selon le procédé décrit ci-dessus, entre noeuds choisis aléatoirement. Chaque noeud peut rejoindre le réseau à l'instant qu'il souhaite, contribuer par un modèle de mélange estimé localement et bénéficier les mélanges estimés circulant à cet instant dans le réseau.

Nous avons observé des résultats expérimentaux intéressants, dans la mesure où les mélanges circulant sur le réseau sont rapidement proches les uns des autres, et généralement meilleurs que tous les modèles locaux initiaux. Nous n'avons pas encore, autant que souhaitable, établi de propriétés théoriques sur la convergence.

Par ailleurs, nous avons inséré, après chaque opération d'agrégation, une phase de validation du mélange réduit sur des données d'un noeud tiers choisi aléatoirement. Cet ajout augmente la robustesse de l'estimation collective à des modèles contributeurs de médiocre qualité.

Plus récemment, nous avons élaboré une version un peu différente de cette proposition [18], étendant un résultat présenté dans (Vasconcelos & Lippman 1998). Reprenant l'esprit de l'algorithme précédent consistant à trouver des regroupements de composantes, on réalise, cette fois, un algorithme EM (plutôt que CM) opérant sur les composantes (chaque composante du mélange réduit est donc une combinaison linéaire de toutes les composantes du mélange redondant). Comme dans la technique précédente, le mélange redondant n'intervient que via ses paramètres et on n'a pas recours aux données. Comme l'objectif est de supprimer la redondance dans le mélange redondant, le critère du maximum de vraisemblance n'est pas satisfaisant. Nous lui préférons une estimation bayesienne du mélange réduit, qui va permettre de réaliser naturellement l'élimination des composantes inutiles dans le mélange. Le mécanisme employé conférent cette bonne propriété est une distribution a priori de Dirichlet,

[18] ce travail n'est pas encore publié.

qui favorise la mise à zéro des poids des composantes peu utiles. Cette étape d'élimination est insérée dans la boucle d'un algorithme itératif de résolution variationnelle appliquée aux mélanges de lois. Nous avons, pour cela, adapté cette technique de résolution de mélange de lois (Attias 1999, Bishop 2006) ; nous substituons, aux données, les paramètres du mélange redondant. Le coût calculatoire de cette technique est, en principe, avantageux, relativement à la proposition précédente, parce qu'il n'est pas nécessaire d'essayer plusieurs hypothèses de complexité de modèle. La qualité des résultats reste à évaluer plus précisement.

Le mécanisme bayesien a été ici mis à profit essentiellement pour mieux traiter la détermination du nombre de composantes dans le modèle réduit ; les distributions a priori (lois conjuguées classiques : gaussienne/Wishart/Dirichlet) dans une estimation sont identiques pour chaque étape de propagation par rumeur. Une perspective intéressante de cette démarche serait de propager les incertitudes sur les paramètres à travers le processus de rumeur.

4.4 Index sur modèles probabilistes

Nous décrivons ici un travail réalisé dans le cadre de la thèse de J.Rougui (co-tutelle franco-marocaine, co-encadrée avec J.Martinez, D.Aboutajdine et M.Rziza). La thèse, dans son ensemble, consiste en un système de recherche d'information dans des documents radiophoniques parlé. Le travail présenté fait partie de ma contribution à ce co-encadrement.

Le travail est suscité par le besoin de structures d'index adaptées pour pouvoir interroger efficacement un ensemble de données multimédia. Une voie classique en bases de données consiste en des structures arboresentes : dans la phase d'indexation, on regroupe les données similaires en paquets, décrits de manière concise. Dans la phase d'interrogation, plutôt que recourir exhaustivement aux données, on interroge d'abord les paquets, pour n'ensuite interroger que la sous-partie de la base la plus pertinente. Dans le cas d'images décrites par des vecteurs de grande dimension, les problèmes liés à la recherche par similarité ont été étudiés par ex. dans (Berrani, Amsaleg & Gros 2002). Une autre voie, que nous avons choisie, s'intéresse à indexer un ensemble de classes, chaque classe étant supposée décrite par sa distribution de probabilité conditionnelle à la classe[19] . Dans le contexte, étant confronté à des données "requête" supposées issues de l'une des classes, on s'intéresse à un moyen d'éviter de calculer la vraisemblance de ces données pour l'ensemble des classes candidates.

En supposant un ensemble de classes, chaque classe étant décrite par un mélange gaussien, nous avons exploré la possibilité de former des "paquets" de classes pour permettre, comme évoqué plus haut, une stratégie d'élagage de l'espace de recherche lors de l'interrogation. Pour la réalisation expérimentale, nous encore avons pris la tâche de reconnaissance du locuteur.

La stratégie de construction d'index sur des mélanges de gaussiennes dépend du degré de liberté donné dans l'estimation des paramètres. En effet, on peut parfois imposer une structure de mélange commune à l'ensemble des classes, ce qui permet de travailler ensuite dans l'espace des paramètres des modèles, plutôt que sur des lois de probabilité (Mami & Charlet

[19] Cette section décrit, dans ses grandes lignes, le travail publié dans J.E. Rougui, M. Gelgon, D. Aboutajdine, N. Mouaddib, M. Rziza, *Organizing Gaussian mixture models into a tree for scaling up speaker retrieval*, Pattern Recognition Letters (Elsevier), vol 28, pp. 1314-1319, 2007, inséré à la fin de ce chapitre

2002). Un mécanisme efficace pour y parvenir consiste à réaliser une estimation bayesienne du mélange où la distribution a priori, estimée sur une cohorte, est commune à toutes les classes (Bimbot, Bonastre, Fredouille, Gravier, Magrin-Chagnolleau, Meignier, Merlin, Ortega-Garcia, Petrovska-Delacretaz & Reynolds 2004, Sturim, Reynolds, Singer & Campbell 2001). Nous avons abordé l'hypothèse alternative, où on ne peut pas maîtriser le processus d'estimation des mélanges : des distributions de probabilité similaires peuvent alors être décrites par jeux de paramètres assez différents, et parfois des nombres de composantes différents. Dans ce cas, il est délicat de travailler dans l'espace multivarié des paramètres des modèles, où la proximité entre vecteurs de paramètres ne reflète pas toujours la similarité entre les lois correspondantes.

Ce travail exploite la même approximation de la divergence de Kullback entre mélanges gaussiens que le travail précédent, qui permet leur comparaison et leur agrégation de mélanges. Nous avons proposé son utilisation pour grouper des mélanges en "paquets" et construire une représentation de chaque paquet, par application de la technique de réduction de modèle. Vis-à-vis de critères de type "vraisemblance croisée", cette manière de construire un arbre est bien plus économique, puisqu'elle ne requiert pas un accès aux données. Diverses stratégies ont été mises en oeuvre et comparées expérimentalement pour le regroupement : une classification hiérarchique ascendante et un algorithme de type "k-means" ; dans les deux cas, les entités de base sont des mélanges.

L'élagage de l'espace de recherche par cet index donne des résultats expérimentaux encourageants, mais sur un corpus encore limité. Par ailleurs, il en résulte une perte de fiabilité de la classification qu'il nous reste à caractériser théoriquement, de manière à rester maître du compromis entre l'élagage de l'arbre et le gain en coût de calcul.

4.5 Conclusion

Ce chapitre a présenté des premiers travaux reliant l'estimation de modèle probabiliste et le contexte des bases de données et systèmes répartis. La voie que nous pensons privilégier pour les quelques années à venir est la classification de données multimédia en contexte réparti, pour les raisons énoncées plus tôt dans ce chapitre. Il s'agirait d'abord d'examiner plus profondément le cas des mélanges gaussiens. L'approche variationnelle pour réduire la redondance reste d'abord à valider complètement. Ensuite, l'aggrégation des mélanges étant actuellement un peu "naïve", on pourrait envisager une propagation plus riche des incertitudes associées aux estimations, dans le processus de rumeur. De manière plus lointaine, on pourrait fonctionner de manière semi-supervisée, voire imaginer des stratégies de placement des données sur des noeuds, qui tireraient parti de l'apprentissage collectif que nous réalisons.

De manière annexe, dans la thèse d'A. Nikseresht, nous avons examiné, même si le travail est resté en suspens, la possibilité de substituer, aux mélanges gaussiens, des fonctions noyaux discriminant deux classes, de manière probabiliste. La technique utilisée est une "Relevance Vector Machine" (Tipping 2002) qui, parce que le procédé de choix des noyaux incorpore un mécanisme bayesien de parcimonie, semble bien se prêter à l'aggrégation de modèles discriminants. Il reste à voir dans quelle mesure on peut ne travailler qu'au niveau des paramètres des noyaux, plutôt que revenir aux données.

Concernant le problème de construction d'arbres de mélanges gaussien, nous étudions actuellement la question suivante. Dans le cas classique, la quantité essentielle pour l'interrogation est la vraisemblance des données à classer, conditionnellement à chaque modèle candidat. Un intérêt de la technique proposée est que nous disposons, en chaque "paquet" et pour chaque feuille associée, d'une estimation de la perte de log-vraisemblance induite par l'utilisation de l'index. Cette estimation est directement liée à ce que la construction des paquets minimise la divergence de Kullback (ou du moins, son approximation) entre le modèle représentant d'un paquet et les modèles des feuilles associées. A cette fin, on pourrait employer une approximation récente de la divergence de Kullback entre mélanges gaussiens (Goldberger & Aronowitz 2005), plus précise que cette que nous avons employée, et pour laquelle nous cherchons actuellement construire un algorithme de réduction de mélange, analogue à celui que nous avons employé jusqu'ici.

4.6 Sélection de publications pour ce chapitre

Les noms des jeunes chercheurs que j'ai encadrés sont soulignés.

1. A.Nikseresht, M. Gelgon, *Gossip-based computation of a Gaussian mixture model for distributed multimedia indexing*, accepté pour publication en sep. 2007 dans IEEE Transactions on Multimedia.

   ```
   >De : aizawa@hal.t.u-tokyo.ac.jp
   >Sep 17, 2007
   >Dr. Marc GELGON
   >Nantes university
   >rue C.Pauc
   >Nantes, 44306
   >Paper:MM001325.R1 Gossip-based computation of a Gaussian mixture
   >model for distributed multimedia indexing
   >Dear Dr. GELGON,
   >We are pleased to inform you that the above paper has been ACCEPTED
   > as a regular paper in the IEEE Transactions on Multimedia.
   (après un cycle de révision mineure)
   ```

2. J.E. Rougui, M. Gelgon, D. Aboutajdine, N. Mouaddib, M. Rziza, *Organizing Gaussian mixture models into a tree for scaling up speaker retrieval*, Pattern Recognition Letters (Elsevier), vol 28, pp. 1314-1319, 2007

Gossip-based computation of a Gaussian mixture model for distributed multimedia indexing

Afshin Nikseresht and Marc Gelgon

Abstract— **The present paper deals with pattern recognition in a distributed computing context of the peer-to-peer type, that should be more and more interesting for multimedia data indexing and retrieval. Our goal is estimating of class-conditional probability densities, that take the form of Gaussian mixture models (GMM). Originally, we propagate GMMs in a decentralized fashion (gossip) in a network, and aggregate GMMs from various sources, through a technique that only involves little computation and that makes parcimonious usage of the network resource, as model parameters rather than data are transmitted. The aggregation is based on iterative optimization of an approximation of a KL divergence allowing closed-form computation between mixture models. Experimental results demonstrate the scheme to the case of speaker recognition.**

I. Introduction

The technical issue addressed by this paper is the distributed computation of a probability density, while the applicative motivation is multimedia document indexing, in the particular context of a decentralised, distributed system. In this section, we first argue for a vision, towards which the scheme we disclose afterwards only supplies a small brick, but we believe this foreword to be both stimulating and necessary to justify the applicative relevance and some technical aspects of the proposal.

A central and classical need expressed by content-based indexing of multimedia documents is the assignment of a symbolic class label to a document or portion thereof, such as identifying a face, a speaker or a spatio-temporal texture or event [9]. Supervised learning is the general task for inducing the class of unlabeled data from labeled examples. Building a search engine able to recognize very many kinds of such audiovisual classes is a formidable task, long rated as unrealistic by the computer vision community, that is currently reviving as one of the most stimulating visions for both research and applications in the field [18], the other major work direction being ability to find different very different views of the same physical scene [3]. Characterizing classes involves a careful design of media-specific observations from the raw data, but by and large, there should all the more features as there are more classes to be distinguished, which in turn increases the amount for training data needed and the computation power required. Briefly stated, the work direction is promising but very demanding.

There are, however, encouraging trends towards the perspective of automatic large-scale harvesting of training examples : (i) joint text/image analysis, which may be fed by a massive resource of web pages, (ii) recent advances in weakly supervised learning [20], [23], which enables learning a class from instances supplied in clutter (e.g. a face within a complex background), and semi-supervised learning [7], which can handle jointly class-labelled and unlabelled data in the training phase. This suggests that the (necessary huge) amount of training data would inherently be distributed on a large scale and provided by independent sources that may join or leave the network. Both for alleviating the computational cost of learning and for reducing the amount of data on the network, we examine the case where supervised learning itself is distributed and, more precisely, decentralized. A suitable organization for the above vision is a peer-to-peer architecture [15] which nodes would run a service providing supervised learning of a multimedia class and would possibly store some training data. Upon request, it could classify incoming data to the best of its current knowledge. A peer-to-peer organization of participant nodes seems relevant, since (i) resources are dynamic : data and learning/classification services can join or leave the network at any time; (ii) a node is both client and server : nodes can learn from one another; (iii) resources are aggregated : the quality of the global service is due to its collective aspect; (iv) the system is decentralized : each contributor can supply data or learning tools, without any central administration. Similar ideas are also being examined for collective learning from text data [21] and sensor networks [17].

As this is a broad perspective, we now restrict the paper to decentralized supervised learning of a class. We do not address herein important issues such as service and data localization, elaborate data placement schemes (examined in [16] for retrieval of similar images), the fact that class identifiers should conform to a standard, nor the query phase.

Let us consider *statistical supervised learning of a class*. To allow for a flexible evolution of the set of classes, we favour a generative approach, that characterizes the class in feature space, over a discriminant approach, that learns directly to distinguish it from other classes. This generative approach leads to more tractable solutions, as introduction of new classes into the system does not require any update to description

The authors are with Laboratoire d'Informatique Nantes-Atlantique (LINA FRE CNRS 2729) and INRIA Atlas project-team, Polytech'Nantes/Université de Nantes, France. Email : firstname.lastname@univ-nantes.fr

This work was for part funded by Region Pays-de-la-Loire (MILES project), and SFERE.

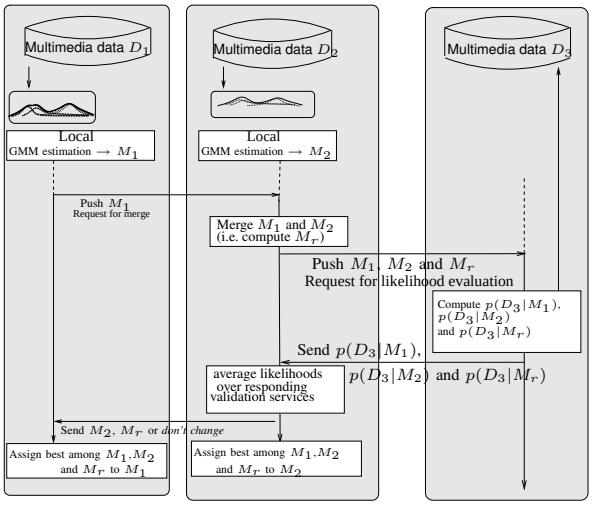


Fig. 1. This figure illustrates a single gossip cycle. Let us consider 3 nodes (the three columns of the figure) that all aim at estimating the class-conditional density of the same class. Each owns some training data, on which it computes it local estimate, of which only M_1 and M_2 are relevant in this figure. M_1 and M_2 are merged into M_r, the quality of which of evaluated over the data D_3 of the third node. The best performing model is then assigned to nodes 1 and 2.

of known classes. Consequently, the remainder of this paper will describe the technique for a single class. Practically, *we estimate the class-conditional probability density*. In this paper, the feature space is assumed common to all nodes in the network. While this leaves space for extending the work, this assumption does not contradict the ideas of the proposal and is applicable to the speaker recognition task on which we apply it.

We further focus on the *case where all densities are Gaussian mixture models*. This model form is of ubiquitous use in modelling multimedia data, for it has numerous good properties (density modelling accuracy, good behaviour in high dimension space, clean procedures for estimation and model complexity determination). They have widely been used to model audio classes [19], images [11] or motion-based spatio-temporal events in videos [9].

The mechanism we employ to propagate mixtures between cooperating nodes that participate in the scheme is *gossip*. Algorithm 1 defines its simplest version for our problem. Gossiping, here, is a non-ending background process in which acquainted nodes may share their models. Any node may then supply, at any time, an estimate of the model ; this estimate improves over time, thanks to the mechanisms proposed below.

Fig. 1 shows the procedure for a single gossip cycle. In this work, the distribution of computation and data is due to the applicative context, in which independent systems cooperate. The key goal of the system is to obtain an estimate which quality is close to what would have been estimated in a centralized version.

Despite its simplicity, this asynchronous, decentralized technique is very effective. Its good properties are extensively reviewed in [8], but may summed up for our problem as follows :

- *speed up* by implementing coarse-grain parallelism over the set of nodes. This occurs at two levels : (1) gossip-based parallelism of learning by merging (step 2 of Algorithm 1) and (2), for each step of (1), parallelism in the computation of the likelihoods for validation (step 3 of Algorithm 1);
- *robustness* both in the distributing computing and statistical estimation senses, since :
  - any node may *leave* during the gossiping without causing major degradation or *join* and obtain, with high probability, an effective estimate of what has been previously collectively estimated on the network,
  - a very poor estimate in a minority of nodes does not affect the collectively estimated model.

Efficiency of the proposed technique comes the two main following features:

- merging density estimates between nodes only involves transmission of, and computation on, mixture model parameters, rather than the generally large amount of multimedia data (or feature vectors that represent it). As a result:
  - the amount of information to be sent on the network is very low ;
  - computation on nodes remains low, relatively to estimation tasks that operate on the multimedia data or feature vectors,
- during the gossip-based model learning phase, the complexity of any mixture (i.e. the number of Gaussian components) keeps a constant order of magnitude. Let us underline that the distributed learning phase and the querying phase, can fully overlap, since mixture reduction keeps the class representation directly ready for query evaluation.

The key mechanism that enables these properties is a criteria and an algorithm related to merging two (or more) mixture models, which are exposed further down.

A work which goal is close to ours, i.e. gossip-based distributed estimation of the parameters of a Gaussian mixture, has been recently presented in [13]. Their approach consists in introducing parallelism in the EM algorithm, by gossiping the M-step, resorting to original data. In our case, in contrast, each contributing node is in charge of estimating its local Gaussian mixture model, and is free to use any mixture model parameter estimation technique for this. The latter point gives an interesting degree of freedom towards a completely decentralized system : only the mixture description need to be standardized, while the node may benefit from recent advances in mixture estimation techniques (e.g. variational Bayes [2], or versions suitable for large amounts of data [22]). Further, the averaging in [13] between the parameters to be merged is simply uniform. To our understanding, a more central difference is that their way of merging knowledge between mixture models does not (at least explicitely) address

Algorithm 1 A gossip cycle for merging-sharing Gaussian mixture models

1. Select at random two nodes in the network, which models are M_1 et M_2 (practically, nodes should autonomously select their partners in a dialogue)
2. Concatenate the components of M_1 and of M_2 to form a single model M_c, then reduce M_c to a merged model M_r with lower number of Gaussian components.
3. Evaluate which among M_1, M_2 and M_r better describes third party data from the data class. A key point is that generally, M_r will perform best.
4. Assign this best model to M_1 and to M_2

correspondence between components to be merged, and leaves open the issue of merging models with different number of components. More generally, we shall see that our technique is amenable to variation of the number of components in the mixture along the gossiping process.

In the remainder of this paper, we detail the proposed approach for distributed model learning (section 2) and demonstrate it on the example of a speaker recognition task (section 3). Section 4 provides concluding remarks.

II. Parameter-level merging of Gaussian mixture models

This section justifies and details how mixture models may be merged using parameter-level rather than data-level computations : section 2.1 defines the optimality criterion aimed at of the merged model, while section 2.2 discusses an approach for conducting the corresponding optimization.

A. Optimality criterion

Let two nodes each carry different probabilistic Gaussian mixture models, denoted $M_1(x)$ and $M_2(x)$, associated to the same multimedia class and hence hidden density $p(x)$. The mixtures can be expressed as :

$$M_k(x) = \sum_{i=1}^{m_k} w_k^i N_k^i(x), \qquad k = 1, 2 \tag{1}$$

where $N_k^i(x)$ is a Gaussian component which mean is μ_k^i and covariance Σ_k^i and the w_k^i are scalar weights. Model M_k is estimated on a data set of size n_k located on node k. $p(x)$ can be estimated by concatenating incoming mixtures as follows :

$$M_c(x) = \frac{1}{n_1 + n_2}(n_1 \sum_{i=1}^{m_1} w_1^i N_1^i(x) + n_2 \sum_{i=1}^{m_2} w_2^i N_2^i(x)) \tag{2}$$

However, the $m_1 + m_2$ components in M_c are generally largely redundant, which implies a useless increase in evaluation cost of likelihoods for this density at query time, when merges are chained gy gossip. Consequently, scaling up the scheme requires transforming M_c into a reduced mixture $M_r = \sum_{i=1}^{m_r} w_r^i N_r^i(x)$ that preserves reasonnably well the density while only having the necessary number of components for this. The point of this policy is that the order of magnitude of the number of components is kept constant through propagation, although it may fluctuate to fit the complexity of the density.

The class models in the nodes would be used to classify new data, typically based on maximum likelihood or more elaborate criteria involving the likelihood. In order to preserve the likelihood as much as possible, we seek a mixture model M_r which maximizes the expected log-likelihood of data D assumed to be drawn from M_c, see (3). It is classically established [5] that this amounts to minimizing the Kullback-Leibler divergence $KL(M_c \| M_r)$, defined by (5), which, in short, measures the loss of information due to the approximation of M_c by M_r :

$$\hat{M}_r = \arg\max \quad E_{M_c} \; [\; ln \;\; p(D|M_r) \;] \tag{3}$$

$$\hat{M}_r = \arg\min \quad \left[\; - \int M_c(x) \; ln \; M_r(x) \; dx \; \right] \tag{4}$$

$$\hat{M}_r = \arg\min \quad \left[- \int M_c(x) \; ln \; \frac{M_r(x)}{M_c(x)} \; dx \; \right] \tag{5}$$

A major issue for the practical computation of (5) is the lack of closed form for this divergence, in the case of Gaussian mixtures, but we propose a bypass in the form of the following approximation. Linearity of the integral applied to (4) provides :

$$\hat{M}_r = \arg\min \quad \left[- \sum_i w_c^i \int N_c^i(x) \; ln \; M_r(x) \; dx \right] \tag{6}$$

In each term of the sum in (6), we approximate the mixture M_r by only one of its Gaussian components, selected as the best approximation to N_c^i, in the KL sense. This leads to the following similarity measure :

$$d(M_c, M_r) = \sum_{i=1}^{m_1+m_2} w_c^i \min_{j=1}^{m_r} KL(N_c^i \| N_r^j) \tag{7}$$

This similarity measure can easily be computed at low-cost, since the Kullback divergence between two Gaussians, which parameters are (μ_1, Σ_1) and (μ_2, Σ_2), benefits from the following closed-form expression :

$$\frac{1}{2}(log\frac{|\Sigma_2|}{|\Sigma_1|} + Tr(\Sigma_2^{-1}\Sigma_1) + (\mu_1 - \mu_2)^T \Sigma_2^{-1}(\mu_1 - \mu_2) - \delta) \tag{8}$$

where δ is the dimension of the feature space.

B. Optimization : an iterative scheme and its initialization

To gain insight into complexity, we assume $m = m_1 \approx m_2 \approx m_r$. The search space is of size $O(m^2)$ and typically cannot be searched exhaustively if there are more than 10 components, which is common when modelling multimedia classes. Hence, we optimize locally criterion (7) with an iterative scheme detailed in Algorithm 2 below, which is adapted (by several aspects) from a technique [10] proposed in the context of hierarchical clustering. The procedure bears analogy

with the classical k-means algorithm, in that it operates local optimization by alternatively assigning elements to groups and re-computing group representatives. In our context, the elements are the components of M_c and the representatives those of M_r.

As often done with k-means, the initial assignements π^0 from which local optimization proceeds could be drawn randomly. Our context suggests a more effective initialization criteria in our context : since generally, Gaussian components coming from the same mixture are not redundant, we draw π^0 at random with the constraint that components arising from the same mixture are not initially grouped. The iterative scheme may still regroup them later, if the data drives it that way. As we draw multiple starting points to retain the best local optimum, this strategy improves sampling of the search space.

C. Complexity of the reduced model

An important point in the proposed approach is the determination of the number of Gaussian components in the reduced model M_r. The seminal study reported in [1] showed that estimating the Kullback divergence is in fact affected by a bias that grows with the number of parameters to be estimated, i.e. with the number of components. It also supplies a first-order approximation of this correction, which we apply here to the definition of $d(M_c, M_r)$, which hence becomes :

$$d(M_c, M_r) = \sum_{i=1}^{m_1+m_2} w_c^i \min_{j=1}^{m_r} KL(N_c^i \| N_r^j) \quad + \quad \nu_{M_r} \quad (13)$$

where ν_{M_r} is the number of independent parameters in the mixture. Our experimental results back the application of this approximation : the number of components obtained in practice appears very similar to that obtained by usual (AIC,BIC) model selection criteria on the model computed directly on all the data (i.e. discarding the distributed aspect of the learning process). We evaluate exhaustively from 1 to m_1+m_2 the performance of each possible number of components in M_r, in independent trials. A faster alternative would be to compute this recursively downwards from $m_1 + m_2$ to 1, but experimental results suggest this can excessily prune the search space at early stages.

D. Validation of a merge operation

In this section, we discuss the need to validate the ability of M_r to generalize to the complete data over the network. Generally speaking, estimation of statistics by gossiping may be shown to converge in some cases (e.g. computing a means and quantiles [12]) but in our problem, the lack of a global view on the data occasionnally leads to a situation where $\hat{M}_r$ is a better model than M_1 and M_2 for local data $D_1 \cup D_2$, but worse on the complete data over the network.

We thus introduce in the scheme a step to validate M_r on third-party data. As described in algorithm 3, it consists in sending $\hat{M}_r, M_1$ and M_2 to a sufficient number of randomly selected acquaintance nodes, each of which make these models compete on its local data and returns the corresponding three likelihoods. When the requesting node has received a sufficient number of such responses (4 in our experiments, further work could make this adaptive), it takes the decision to validate M_r or reverse to M_1 and M_2.

This phase loads the network with more messages, but (i) these messages are very short and no multimedia data nor feature vectors are transmitted (ii) computation of likelihoods is inexpensive.

While exchanges between nodes in this validation step may be implemented in a variety a ways, limited depth network flooding is an interesting one, offering the following perspective of extension on the present work : likelihood information being collected and aggregated through the flooding may be useful not only to the node emitting the request, but also to other nodes, since at no extra cost, they can learn from it about the quality of their own model, relatively to others. In other words, a ranking of the nodes may be learned in a decentralized way, which could help route queries to more effective models.

III. Experimental results

The example of distributed speaker recognition is taken throughout this section, as it is a representative case where Gaussian mixtures are very popular. The technique however directly applies to a wide range of audiovisual classes. We first focus on the merging operation, i.e. at local scale (section III-A). We then observe global performance, in the context of gossip-based mixture propagation (section III-B). Throughout these experimental results, the figure of merit is the quality of the class-conditional pdf estimate, in particular with respect to a conventional, centralized approach, rather than ability of the scheme to classify new data correctly. The latter however derives direcly from the former in a Bayesian decision rule.

A. Detailled view on one or two merge operations

In the first experiment, each of three nodes has learnt a class-conditional density for a speaker in a common 13-dimension mel-cepstral feature space [19]. The three corresponding mixtures merge simultaneously into a single mixture (straightforward since (1) generalizes to merging more than two mixtures). Each node was provided with different training data from the same speaker and the duration of audio recordings was between 7 to 16 seconds, depending on node. Each node provides a mixture estimate from local data, and is free to choose the precise technique used for this. For our experiments, the Expectation-Maximization local optimization algorithm is employed, with some enhancements [4] to limit poor local minima. Each mixture also autonomously and automatically determines its number of components (in practice, the common BIC criterion was used). All covariance matrices in the mixture are full (rather than spherical or diagonal).

The three incoming nodes respectively have 4,4 and 5 components. Their concatenation into M_c supplies a 13-component model, which should be reduced to a lower number of components, to be determined. Fig. 2(a) displays, in the example case of the second feature vector, the three incoming densities, the concatenated density and the density after mixture reduction.

Algorithm 2 Iterative optimization algorithm for estimating the reduced model M_r (criterion (7))

for m_r : from 1 to m1+m2 **do**
Start from a constrained random initialization $\hat{\pi}^0$ (or given, if available)
$it = 0$
repeat
1. Re-fit mixture M_r :
given the current component clustering $\hat{\pi}^{it}$, set initially or computed at the previous iteration, update mixture model parameters as follows :

$$\hat{M_r}^{it} = \arg \min_{M_r \in \mathcal{M}_{m_r}} d(M_c, M_r, \hat{\pi}^{it}) \tag{9}$$

where $\mathcal{M}_{m_r}$ is the space of all mixture with m_r components that may be formed by grouping components of M_c. This re-estimation in fact amounts to updating each component of M_r as follows. For component j, algebra leads to the following expressions :

$$\hat{w}_r^j = \sum_{i \in \pi^{-1}(j)} w_c^i, \quad \hat{\mu}_r^j = \frac{\sum_{i \in \pi^{-1}(j)} w_c^i \mu_c^i}{\hat{w}_r^j}, \quad \hat{\Sigma}_r^j = \frac{\sum_{i \in \pi^{-1}(j)} w_c^i (\Sigma_c^i + (\mu_c^i - \hat{\mu}_r^j)(\mu_c^i - \hat{\mu}_r^j)^T)}{\hat{w}_r^j} \tag{10}$$

where $\pi^{-1}(j)$ is a light notation for $\hat{\pi}^{-1,it}(j)$, the set of M_c that project onto component j in M_r. Let us note that $\hat{\Sigma}_r^j$ is generally non-diagonal, even if the components being grouped have diagonal covariance matrices, such as is often the case with decorrelated features used in e.g. speech or speaker recognition.
2. Grouping components :
for mixture $\hat{M}_r^{it}$ obtained in Step 1, we seek the mapping π^{it+1}, defined from $\{1, \ldots, m_1 + m_2\}$ into $\{1, \ldots, m_r\}$, which best groups components of M_c to build components of $\hat{M}_r^{it}$, in the following sense :

$$\hat{\pi}^{it+1} = \arg\min_{\pi} \; d(M_c, \hat{M}_r, \pi) \tag{11}$$

In other words, each component i of M_c projects onto the closest component j of $\hat{M}_r^{it}$, according to their Kullback divergence ((12) below). In this phase, we resort to exhaustive search among 'source' components, which has a low-cost, thanks to the availability of (8).

$$\pi^{it+1}(i) = \arg\min_{j} KL(N_c^i || N_r^j) \tag{12}$$

3.*it=it+1*
until convergence (i.e. $\pi^{it+1} = \pi^{it}$)
compute $d(M_c, \hat{M}_r) = \sum_{i=1}^{m_1+m_2} w_i \min_{j=1}^{m_r} KL(N_c^i \| N_r^j) \quad + \quad \nu_{M_r}$
end for
Retain model $\hat{M}_r$ which minimizes $d(M_c, \hat{M}_r)$ over the set of candidate mixture complexities explored.

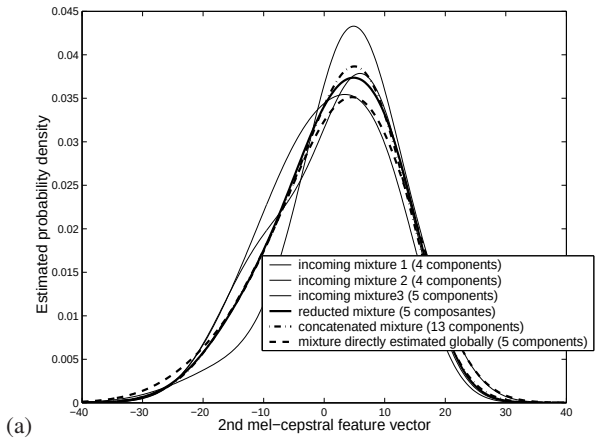
Algorithm 3 Validation of a merge

for enough times **do**
Draw node k at random among acquaintance nodes
Sends $\hat{M}_r, M_1, M_2$ to node k running a GMM evaluation service
Node k computes $p(D_k|\hat{M}_r), p(D_k|M_1), p(D_k|M_2)$ and sends them back to current node.
end for
if $p(D_k|\hat{M}_r) > p(D_k|M_1)$ and $p(D_k|\hat{M}_r) > p(D_k|M_2)$ **then**
Validate the merge operation (proceed as in Algorithm 1)
else
ignore it and keep M_1 and M_2
end if

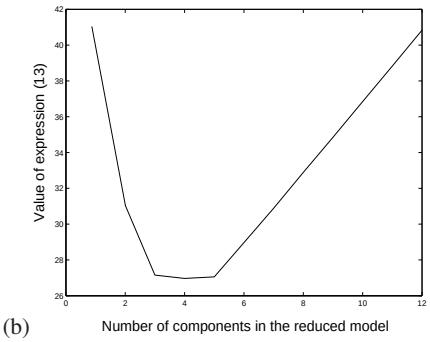
The mixture estimated (again enhanced EM) on the whole the data over the network is also plotted. While the main point of this paper is to propose a decentralized alternative to this, this direct model (denoted M_d) serves herein as a reference density against which we evaluate the loss due to distribution of the data and computations. Fig. 2(b) shows that criterion (13) chooses a reduction from 13 to 4 components. To evaluate the effectiveness of the mixture reduction, Fig. 2(c) provides numerical evidence in terms of Kullback-Leibler loss between reference mixtures (M_d and M_c) and approximating mixtures. KL divergence is used (rather than its approximation proposed in (7)). We evaluate it by a Monte-Carlo procedure with N=10^8 samples, as follows :

$$KL(p,\tilde{p}) \approx \frac{1}{N}\sum_{i=1}^{N} log\frac{p(x)}{\tilde{p}(x)} \qquad (14)$$

where p and $\tilde{p}$ respectively denote an ideal model and its approximation. While this should be closer to the true loss than (7)), its computational cost forbids its usage in the scheme, it is only used here for external assessement.



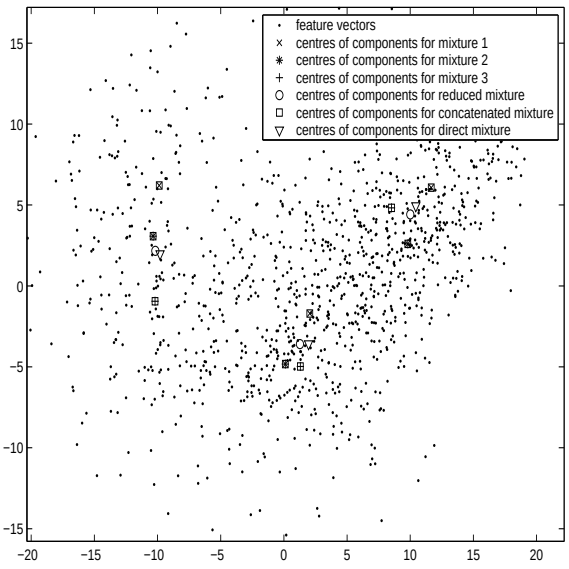
(a)



(b)

(c)

	KL(concatenated ‖...)	KL(direct‖...)
mixture 1	0.0241	0.0306
mixture 2	0.0030	0.0219
mixture 3	0.0101	0.0309
reduced mixture	0.0005	0.0087



(a)

(b)

	KL(concatenated ‖...)	KL(direct‖...)
mixture 1	0.125	0.112
mixture 2	0.109	0.117
mixture 3	0.075	0.082
reduced model	0.015	0.020

Fig. 3. Three mixtures merge (not simulatenously, see main text). (a) shows the feature vectors and the centres of the Gaussian components (for incoming, concatenated, reduced models, as well as, for reference, the mixture that could be directly estimated over the whole data set). (b) Evaluation of the KL loss between reference densities (concatenated, direct) and models coming in and out of the merge operation.

It can be observed that the direct mixture is much better approximated by the reduced mixture than by any of the incoming mixtures. This does not come at the expense of mixture complexity, since the reduced mixture has 4 components, in fact the same is estimated by a BIC criterion for the direct model.

We report a second experiment, applied to a different speaker. It again involves three nodes but, in contrast to the previous experiment, two nodes are merged, and then a third node is merged to their reduced mixture to form a final reduced mixture. The experiment is conducted in a 2-dimension space, for the sake of clarity of fig. III-A(a). Its purpose is more an illustration value than a demonstration of large scale effectiveness. The centres of the incoming mixtures, as well as the centres of the reduced and direct mixtures are superimposed to the feature vectors, and the two latter are clearly very close.

B. Gossip-based estimation

This section reports the performance of the proposed technique in the gossip setting. Each node owns different data from the same speaker and independently estimates its own model. Practically, EM with multiple starts is employed in our experiments for this purpose but, as stated before, other techniques may be used.

We evaluate the capability of each mixture on the network to model data D from the class of interest (here, a speaker) with the classical marginal likelihood [14]. Data D here is the union of the data dispatched over all nodes, which is never gathered when the practical system runs, but is a relevant figure of merit for external observation.

We carry out the practical computation of the marginal likelihood of the data with the BIC criterion :

$$BIC(D|M) = -log\ p(D|\hat{\theta}) + \frac{\nu\ log(\sharp D)}{2} \qquad (15)$$

where mixture M is defined by a parameter vector θ, $p(D|\theta)$ is the likehood of the data for this model, ν is the number of independent parameters in the mixture and $\sharp D$ the size of the data set (the data set does not need to propagate in the network, but its size should propagate and cumulate in n_1 and n_2)

Fig. 4 depicts, after each gossip cycle and on each node, the evolution of criterion (15), which should be minimized. The following observations can be made. The process stabilizes around a "collective model". Convergence cannot be established, as illustrated in the zoom into Fig. 4, due to the lack of an optimization criterion global to the network, which is the case in the prototypal example of computation of a mean [13], [6]. From a practical viewpoint, however, *all nodes are rapidely assigned a mixture that is better (slightly or largely) than any of the original mixture, which later implies improvement in recognition rates when the system is queried*. In this experiment, the effectiveness of the collective model is significantly better than that of a single mixture model that could have been estimated directly on the whole data (the performance of which is represented by a dashed horizonal line). This latter advantage however reduces when the size of the feature space is large compared to the amount of training data (dimensionality curse). Overall, however, this example, which is representative of many other obtained, suggests that the proposed scheme provides promising results on three points : quality in model estimation, the flexibility of a decentralized system, and speed up thanks to parallel computing.

It should also be underlined that the horizontal axis only indicates time order and is non-linearly related to time, since gossiping is strongly parallel.

As the result depends on the order in which the nodes are merged and on the random initializations involved in the merging algorithm, we draw in fig. 5 statistics to average out those effects. Fig. 5 indicates how variable the likehood is over the set of nodes (this variability is averaged over 50 independent runs of the complete gossip).Variability decreases very fast ; the transient phase with higher variability (up to the 18th gossip cycle) corresponds to not all the nodes having yet participated in the gossip process.

As illustrated in fig. 6, the scheme can easily handle a node that joins the network. In this example involving 20 nodes, an additional node joins after 50 cycles. Soon after it joins, it benefits from the previous exchanges. Indeed, the amount of data available for mixture estimation ((n_1 and n_2) in eq. (2)) cumulates as gossip progresses.

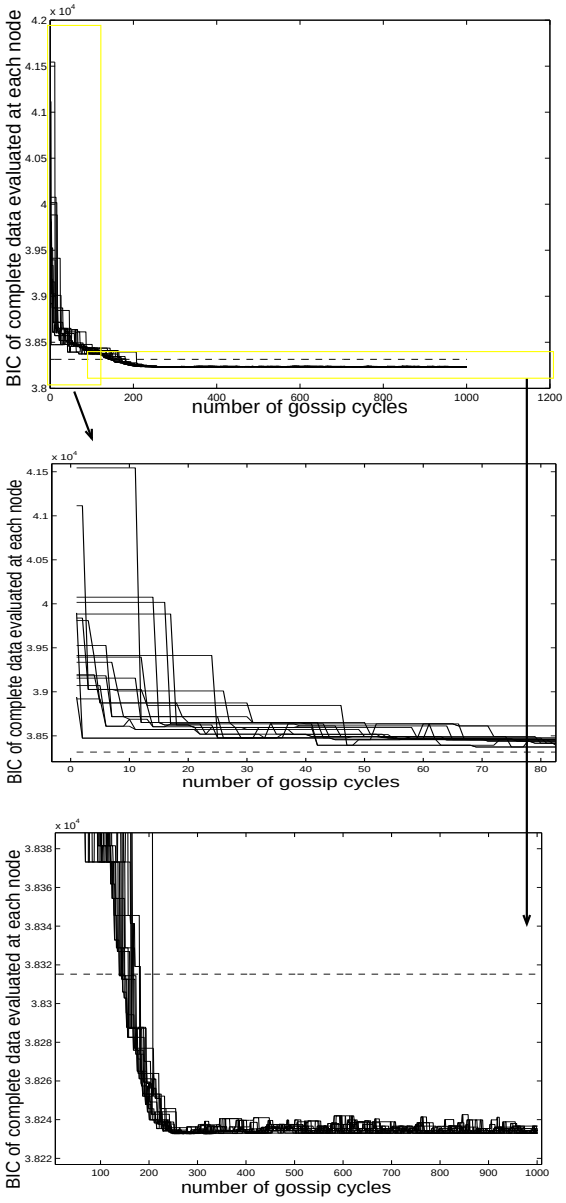


Fig. 4. All graphs measure the BIC criterion over time (this criterion evaluates the ability of the Gaussian mixture models to generalize to all data from the class being learnt; it should be minimized). Top : this evolution is shown for the 20 nodes participating in the experiment. As the 20 graphs are somewhat superimposed, Medium and Bottom figures are zooms on the top figure, for clarity sake.

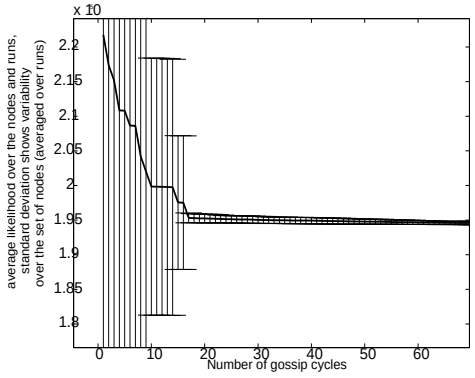


Fig. 5. This graph shows the statistical behaviour of the system : the variability of the likelihood over the set of nodes participating in the gossip. This variability is averaged over 50 runs

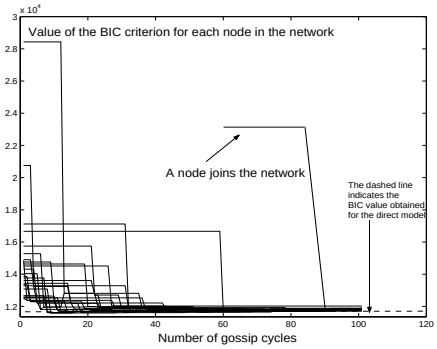


Fig. 6. This experiment illustrates fast integration of a node joining a distributed learning process involving 20 nodes. Right from its first contact (cycle 73), the joining node strongly improves by catching the central trend of the network.

IV. Conclusions

This work fits into a vision towards a multimedia indexing and retrieval system, which would be decentralized and deployed on a large scale. In this setting, algorithmic components are required, that induce low computational cost, incrementality and only require a little amount of bits to transit between nodes.

This paper proposed a novel scheme for this purpose, dedicated to Gaussian mixtures models, which are one of the most useful representations of a multimedia class. The proposal wraps a parcimonous mixture model merging technique into a gossip framework, demonstrating that it can efficiently propagate and collectively improve estimates over time. The point of the gossip framework is that it is well suited to dynamic, decentralized computing environments.

More generally, crossing pattern recognition and large-scale distributed computing is a promising direction in content-based multimedia indexing, since the first ingredient can greatly enhance services offered to users, far beyond file sharing, while the second provides data, computation and algorithmic resources.

References

[1] H. Akaike. A new look at the statistical model identification problem. *IEEE Trans. on Automatic Control*, (19):716–723, 1974.

[2] H. Attias. A variational Bayesian framework for graphical models. In *Neural Information Processing Systems (NIPS) Conference*, Denver, USA, November 1999. MIT Press.

[3] S.A. Berrani, L. Amsaleg, and P. Gros. Robust content-based image searches for copyright protection. In *Proc. of ACM workshop on Multimedia databases*, pages 70–77, New Orleans, USA, November 2003.

[4] C. Biernacki, G. Celeux, and G. Govaert. Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models. *Computational Statistics and Data Analysis*, 41:561–575, 2003.

[5] C. Bishop. *Neural networks for Pattern Recognition*. Oxford University Press, 1995.

[6] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Trans. on Information theory*, June 2006.

[7] F. Cozman, M. Cirelo, T.S. Huang, I. Cohen, and N. Sebe. Semisupervised learning of classifiers : theory, algorithms and their applications to human-computer interaction. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(12):1553–1567, December 2004.

[8] P.T. Eugster, R. Guerraoui, A.-M. Kermarrec, and L. Massoulié. From epidemics to distributed computing. *IEEE Computer*, 37(5), 2003.

[9] R. Fablet, P. Bouthemy, and P. Perez. Non parametric motion characterization using causal probabilistic models for video indexing and retrieval. *IEEE Trans. on Image Processing*, 11(4):393–407, April 2001.

[10] J. Goldberger and S. Roweis. Hierarchical clustering of a mixture model. In *Proc. of Neural Information Processing Systems (NIPS'2004)*, pages 505–512, 2004.

[11] R. Hammoud and R. Mohr. Gaussian mixture densities for video object recognition. In *Proc. on Int. Conf. on Pattern Recognition (ICPR'2000)*, pages 71–75, Barcelona, Spain, August 2000.

[12] D. Kempe, A. Dobra, and J. Gehrke. Gossip-based computation of aggregate information. In *IEEE symp. on foundations of computer science*, Cambridge, MA, USA, October 2003.

[13] W. Kowalczyk and N. Vlassis. Newscast EM. In MIT Press, editor, *Proc. of Neural Information Processing Systems (NIPS) 17*, 2005.

[14] D. MacKay. Bayesian interpolation. *Neural computation*, 4(3):415–447, 1992.

[15] D. Milojicic, V. Kalogeraki, R. Lukose, L. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu. Peer-to-peer computing. Technical Report 2002-57, Hewlett-Packard, 2002.

[16] W.T. Muller, M. Eisenhardt, and A. Henrich. Efficient content-based P2P image retrieval using peer content descriptions. In *Proc. of SPIE Internet Imaging V*, volume 5304, pages 57–68, December 2003.

[17] R. Nowak. Distributed EM algorithms for density estimation and clustering in sensor networks. *IEEE Trans. on Signal Processing*, 51(8), August 2003.

[18] J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, editors. *Towards category-level object recognition*. Springer, 2006.

[19] D. Reynolds. Speaker identification and verification using gaussian speaker models. *Speech communication*, 17:91–108, 1995.

[20] C. Schmid. Weakly supervised learning of visual models and its application to content-based retrieval. *Int. Journal of Computer Vision*, 1(56):7–16, 2004.

[21] C. Tang, S. Dwarkadas, and Z. Xu. On scaling latent semantic indexing on large peer-to-peer systems. In *Proc. of ACM SIGIR (SIG on Information Retrieval*, pages 145–153, Sheffield, U.K., July 2004.

[22] J.J. Verbeek, J.R.J. Nunnink, and N. Vlassis. Accelerated EM-based clustering of large datasets. *Data Mining and Knowledge Discovery*, 2006. in press.

[23] L. Xie and P. Perez. Slightly supervised learning of part-based appearance models. In *Proc. of IEEE Workshop of learning in computer vision and pattern recognition*, Washington, DC, USA., June 2004.

Organizing Gaussian mixture models into a tree for scaling up speaker retrieval

J.E Rougui [a,b], M. Gelgon [a], D. Aboutajdine [b], N. Mouaddib [a], M. Rziza [b],

[a] *Ecole polytechnique de l'université de Nantes*
INRIA Atlas project-team, LINA (FRE CNRS 2729),
BP 50609 44306 Nantes Cedex 03, France
Tel : +33 2 40 68 32 57, Fax : +33 2 40 68 32 32

[b] *Groupe Signaux Communications et Multimedia*
Faculté des Sciences Rabat-Agdal
4, Av Ibn Battouta, Rabat, Morocco.

Abstract

Numerous pattern recognition tasks set in the probabilistic framework face the following issue : it is expensive to evaluate the likelihood function for test data, when there are given very many candidate probabilistic models for explaining this data. We consider the application of this general and important problem to speaker recognition for indexing and retrieval purposes in radio archives. More precisely, we propose to reduce complexity at query time, by prior organization of speaker models into a hierarchy. This is very classically done for multi-dimensional vectors, but we propose herein a technique for building a hierarchy of probabilistic models, in the case these models take the form of a Gaussian mixture. From a closed-form approximation of Kullback-Leibler divergence between parent and children, an optimality criterion and an optimization technique are derived, from which we propose an efficient approach for building a tree of models, using clustering techniques (dendrogram-based or k-means-like). The proposed scheme is evaluated on real data.

Key words: Multimedia indexing and retrieval, speaker recognition, Gaussian mixture, tree-based indexing structure

1 Context and goal

Enhanced content-based indexing, browsing and retrieval in large amounts of audio documents requires prior temporal structuring of this content and labelling of entities extracted, such as assigning the identity of a speaker to a

temporal segment (a task also known as "speaker diarization"). A considerable amount of work has been put forward in this field, during the past ten years (Bimbot et al., 2004). In this paper, we focus on the task of text-independent speaker recognition, applied to spoken radio archives. The front-end to the contribution is a classical one. We partition the audio stream into speaker-homogeneous segments by detecting changes in speaker turns. Each speaker is characterised by a probability density estimate of its Mel-cepstral feature vectors (MFCC). This density is modelled as a Gaussian mixture model (GMM), as this provides an effective trade-off between ability to describe complex densities and ability to estimate correctly the parameters of this model from a limited amount of training data, which is especially challenging in the relatively high dimension spaces formed by Mel-cepstral coefficients (between 10 and 40, generally).

Ideally, indexing of an audio stream in carried out incrementally. In such a case, the task of speaker matching is encountered at two stages: when two temporally disconnected segments contain the same speaker and should be labelled as such, and when a user formulates a query. The need for incrementality, i.e. the ability to accommodate for new speakers in the database, or refine already enrolled speaker models as new information is made available, affects a design choice of the scheme: we make use of generative models, rather than techniques that discriminate between speakers.

A typical solution to speaker recognition consists in exploring exhaustively the set of the $S_1, \ldots, S_M$ enrolled speaker models and evaluating the likehood of the query data given each candidate model. The point this paper wishes to address here pertains to scaling up such a system to a large number of speakers, by organizing the set of candidate speaker models in the form of a tree, with a view to obtaining a sub-linear (i.e. $< O(M)$) computational cost at evaluation time. Clearly, the matter is to trade a significant speed up against minimal loss recognition accuracy, relatively to exhaustive search.

There exist alternative work directions for reducing cost: cepstral subspaces (Nishida and Ariki, 1998; Zhou and Hansen, 2002; Upendra et al., 2001), anchor models (Mami and Charlet, 2002; Sturim et al., 2001), that express speakers in a basis of reference speakers, or considering only a few dominant Gaussians in the mixture. These approaches propose speeding up by reducing the evaluation per speaker but remain $O(M)$; our work direction is orthogonal and complements it.

The task relates tightly to the classical issue of indexing structures for multi-dimensional data. The database community has put forward a considerable amount of contributions based on a variety of tree structures (Berrani et al., 2003; Zezula et al., 2006). The particularity of the current problem arises from the nature of the entities to index, namely probability distributions, for which

classical indexing structures are inappropriate. Extending such structures to handle probabilistic representations is one of the most important current issues, since it has a major impact on the ability to scale up applications to large amounts of data.

The remainder of this paper is organized as follows. Section 2 provides the following preliminary material: given a set of sibling speaker models and their parent, how do we define the representativity of the parent with respect to its children ? Then, how do we build a parent that possesses an optimal representativity ? Section 3 exploits proposals made above to define several alternative techniques for grouping similar speakers and organizing a set of models into a tree. Section 4 reports experimental results, while we provide concluding remarks in section 5.

2 Child-to-parent relation

Let us consider a set of M enrolled speaker models, i.e. M Gaussian mixture models. The manner by which the parameters of these models are estimated is not central in the present proposal: it may be through conventional EM-based estimation (Bishop, 1995) or, more effectively from limited training data, a point estimate from Bayesian learning with universal background model as a prior density (Ben et al., 2004). It suffices to say here that model k is expressed as:

$$S_k(x) = \sum_{i=1}^{m_k} w_k^i N_k^i(x) \tag{1}$$

where $N_k^i(x)$ is a Gaussian component which mean is μ_k^i and covariance Σ_k^i, while w_k^i are scalar weights.

Let us assume recognition is based on maximum likelihood of the query data D, over the set of M candidate models (the scheme extends directly to maximum a posteriori). Exhaustive maximum likelihood search forms the baseline technique, against which we propose improvement.

We aim at forming a hierarchy of speaker models by grouping the M models bottom-up. To justify the criterion proposed below for parent-child similarity, let us consider the simplest tree, where two speakers S_1 and S_2 are represented by a single father S_{12}, which also takes the form of a Gaussian mixture model. This extends directly to an arbitrary number of children.

The cost reduction at query time, when the tree is explored root-to-leaf, is obtained by computing a single value $p(D|S_{12})$ instead of both $p(D|S_1)$ and $p(D|S_2)$. Consequently, S_{12} should thus be designed so that $p(D|S_{12})$ is as close as possible to both $p(D|S_1)$ and $p(D|S_2)$, in order to keep classification

error as close as possible to that of exhaustive search. The number m_{12} of Gaussian components in S_{12} should also be clearly smaller than $m_1 + m_2$ to ensure computational cost reduction of evaluation.

The next two subsections respectively define (sec. 2.1) an optimality criterion for a parent, given its children, and (sec. 2.2) expose how we optimise this criterion to actually determine the parent model from given children.

2.1 Defining a low-cost, minimal KL loss measure between parent and child

The expected loss in log-likelihood caused by approximating both S_1 and S_2 by S_{12} is expressed as:

$$E_{S_k}\left[\ ln\ p(D|S_k)\ \right]\ -\ E_{S_k}\left[\ ln\ p(D|S_{12})\ \right], \text{where} \quad k = 1, 2 \tag{2}$$

Assuming all candidates are equally probable, the optimal mixture $\widehat{S_{12}}$ minimizing this loss is thus defined as:

$$\widehat{S_{12}}\ =\ \arg\min_{\mathcal{S}}\ \left[\ -\int S_1(x)\ ln\ S_{12}(x)\ dx - \int S_2(x)\ ln\ S_{12}(x)\ dx\right] \tag{3}$$

where integrals span the feature space and $\mathcal{S}$ is the search space, discussed below. This corresponds in fact to minimising the Kullback-Leibler divergence $KL(S_{1+2}\|S_{12})$ (Bishop, 1995), where $S_{1+2}(x)$ designates $\frac{1}{2}(S_1(x)+S_2(x))$, i.e.:

$$\widehat{S_{12}}\ =\ \arg\min_{\mathcal{S}}\ \left[\ -\int S_{1+2}(x)\ ln\ \frac{S_{12}(x)}{S_{1+2}(x)}\ dx\ \right] \tag{4}$$

A major issue for the practical computation of (4) is the lack of closed form for this divergence, in the case of Gaussian mixtures. To avoid expensive Monte-Carlo evaluation (Chen et al., 2005), we propose a closed form through the following approximation. Linearity of the integral applied to (3) provides:

$$\widehat{S_{12}}\ =\ \arg\min_{\mathcal{S}}\left[-\ \sum_{i}^{m_1+m_2} w_{1+2}^i \int N_{1+2}^i(x)\ ln\ S_{12}(x)\ dx\right] \tag{5}$$

In each term of the sum in (5), we approximate the mixture S_{12} by only one of its Gaussian components, selected as the best approximation to N_{1+2}^i, in the KL sense. This leads to the following similarity measure, denoted below KL_m for KL_{modified}, between a reference model S_{1+2}, which contains too many components to be efficient, and its approximation S_{12}:

$$\widehat{S_{12}} = \arg\min_{\mathcal{S}} \; [KL_m(S_{1+2}\|S_{12})]$$
$$= \arg\min_{\mathcal{S}} \left[\sum_{i=1}^{m_1+m_2} w_{1+2}^i \min_{j=1}^{m_{12}} KL(N_{1+2}^i\|N_{12}^j) \right] \quad (6)$$

The following expression is used for comparing a child model and its parent model in the tree:

$$KL_m(S_k\|S_{12}) = \sum_{i=1}^{m_k} w_k^i \min_{j=1}^{m_{12}} KL(N_k^i\|N_{12}^j), \; k=1,2 \quad (7)$$

This similarity measure can easily be computed at low-cost, since the Kullback divergence between two Gaussians, which parameters are (μ_1, Σ_1) and (μ_2, Σ_2), benefits from the following closed-form expression:

$$\frac{1}{2}(log\frac{|\Sigma_2|}{|\Sigma_1|} + Tr(\Sigma_2^{-1}\Sigma_1) + (\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2) - \delta) \quad (8)$$

where δ is the dimension of the feature space. It may be demonstrated (Goldberger and Roweis (2004)) that optimising (6) amounts to finding an optimal discrete mapping π between the m_1+m_2 components of S_1 and the m_{12} $(< m_1+m_2)$ components of S_{12}. This involves reducing the number of components in the mixture S_{1+2} to build S_{12}, while minimizing density distorsion, in the KL_m sense. The search space $\mathcal{S}$ thus consists in all ways of grouping the m_1+m_2 components into m_{12} groups.

2.2 Search for the optimal parent mixture

The search space cannot in practice be searched exhaustively if there are more than 10 components, which we typically encounter. Hence, we optimise locally criterion (7) with an iterative scheme detailed in Algorithm 1 below. It is adapted from a technique proposed by Goldberger and Roweis (2004), in the context of hierarchical clustering of Gaussians (rather than Gaussian mixtures). The procedure bears analogy with the classical k-means algorithm, in that it operates local optimization by alternatively assigning elements to groups and re-computing group representatives. In our context, the elements are the components of S_{1+2} and the representatives those of S_{12}.

As often done with k-means, the initial assignments π^0 from which local optimisation proceeds could be drawn randomly. Our context suggests a more effective initialisation criteria in our context: since generally, Gaussian components coming from the same mixture are not redundant, we draw π^0 at random with the constraint that components arising from the same mixture are not initially grouped. The iterative scheme may still regroup them later, if the data drives it that way. As it is practically desirable to draw multiple starting

points to retain the best local optimum, this strategy improves sampling of the search space.

3 Grouping speaker models

This section applies the child-to-parent relation criteria and optimization technique presented in the previous section to three ways of organizing speaker models into a search tree. Practically, the scope of this paper is restricted to a single intermediate layer between the root and the leaves, and may be viewed as clustering of speaker models.

3.1 Dendrogram-based grouping

We first present a transposition of the most classical data clustering to our problem, namely bottom-up hierarchical clustering, where each leaf is a Gaussian mixture model (see fig. 1):

1. a $M \times M$ similarity matrix is computed between models. Similarity between two mixtures S_1 and S_2 is computed as :

$$KL_m(S_1||S_2) + KL_m(S_2||S_1) \tag{15}$$

2. the two most similar models are grouped and summarized as one (here, not reducing S_{1+2} to S_{12}, to keep a richer representation), and so on until there remain only two nodes. The similarity matrix is updated after each merge operation.
3. the dendrogram-tree obtained is cut (dashed line in fig. 1) so that the number of nodes just above it is close to $log_2(M)$. These nodes inheritate from all their (grand)children, and the corresponding mixture model are determined by optimizing criterion (6). Doing so, a tree with a variable of children is formed, which we use for searching.

As usually with hierarchical clustering, this technique is not incremental and its complexity does not scale up well to large amounts of speakers (it does at evaluation time, hence its interest, but not at tree construction time). Since similarity between models is computed by means of KL_m, it only resorts to model parameters rather than data, and is hence practically fast and usable for a moderate number of speakers.

Algorithm 1 Iterative optimisation algorithm for estimating the reduced model S_{12} (criterion (7))

Start from a constrained random initialisation $\hat{\pi}^0$ (or given, if available)
$it = 0$
repeat
1. Re-fit mixture S_{12}:
given the current component clustering $\hat{\pi}^{it}$, set initially or computed at the previous iteration, update mixture model parameters as follows:

$$\widehat{S_{12}}^{it} = \arg \min_{S_{12} \in \mathcal{S}_{m_{12}}} KL_m(S_{1+2}, S_{12}, \hat{\pi}^{it}) \tag{9}$$

where $\mathcal{S}_{m_{12}}$ is the space of all mixture with m_{12} components that may be formed by grouping components of M_c. This re-estimation in fact amounts to updating each component of S_{12} as follows. For component j, algebra leads to the following expressions:

$$\hat{w}_{12}^j = \sum_{i \in \pi^{-1}(j)} w_{1+2}^i \tag{10}$$

$$\hat{\mu}_{12}^j = \frac{\sum_{i \in \pi^{-1}(j)} w_{1+2}^i \mu_{1+2}^i}{\hat{w}_{12}^j} \tag{11}$$

$$\hat{\Sigma}_{12}^j = \frac{\sum_{i \in \pi^{-1}(j)} w_{1+2}^i (\Sigma_{1+2}^i + (\mu_{1+2}^i - \hat{\mu}_{12}^j)(\mu_{1+2}^i - \hat{\mu}_r^j)^T)}{\hat{w}_{12}^j} \tag{12}$$

where $\pi^{-1}(j)$ is a light notation for $\hat{\pi}^{-1,it}(j)$, the set of S_{1+2} that project onto component j in S_{12}.
2. Grouping components:
for mixture $\widehat{S}_{12}^{it}$ obtained in Step 1, we seek the mapping π^{it+1}, defined from $\{1, \ldots, m_1 + m_2\}$ into $\{1, \ldots, m_{12}\}$, which best groups components of S_{1+2} to build components of $\widehat{S}_{12}^{it}$, in the following sense :

$$\hat{\pi}^{it+1} = \arg \min_{\pi} KL_m(S_{1+2}, \widehat{S_{12}}, \pi) \tag{13}$$

In other words, each component i of S_{1+2} projects onto the closest component j of $\widehat{S}_{12}^{it}$, according to their Kullback divergence ((14) below). In this phase, we resort to exhaustive search among 'source' components, which has a low-cost, thanks to the availability of (8).

$$\pi^{it+1}(i) = \arg \min_j KL(N_{1+2}^i || N_{12}^j) \tag{14}$$

3.*it=it+1*
until convergence (i.e. $\pi^{it+1} = \pi^{it}$)
compute

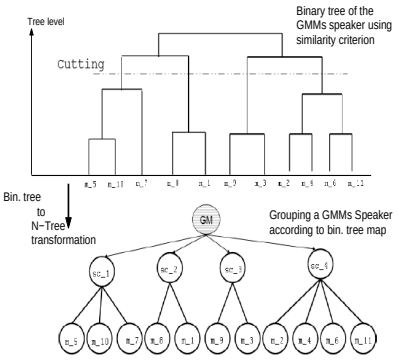


Fig. 1. Hierarchical clustering applied to a set of Gaussian mixture models. (top) A dendrogram is first built, then (bottom) cut to determine nodes (i.e. GMM) forming the intermediate layer

3.2 Iterative grouping

As an alternative to hierarchical clustering, we propose an iterative scheme analogous to the k-means procedure, for which data elements are mixture models. It is detailed in fig.2. The criterion to optimize here generalizes the simple parent-child relation optimality defined in section 2, over several parents :

$$\sum_{S_p \in parents} \quad \sum_{S_c \in \text{ children of } S_p} KL_m(S_p \| S_c) \tag{16}$$

Algorithm 2 Iterative optimisation of parent model parameters

Start from random grouping of speaker models
repeat
1. Re-fit mixture of each parent using Algorithm 1.
This step involves itself a k-means-type algorithm that operates on Gaussian components.
(rather than on Gaussian mixtures as the present Algorithm 2 does).
2. Re-assign each child of the complete set to the most similar parent (in the $KL_m(S_{parent} \| S_{child})$ sense)

until convergence

An essential good property is that assignments of speaker models to groups may easily be questioned, in contrast to dendrogram-based grouping. Consequently, the iterative approach is amenable to incremental processing, i.e. it can accomodate new speaker models at leaves and update the intermediate layer by re-optimizing (16) locally and, if required, it would be quite easy to extend the present scheme to allow the number of intermediate nodes to evolve over time.

3.3 Exploiting the approximation error in the tree structure

Let us consider a tree obtained by either of the approaches presented in the two previous subsections. Let S_p denote a parent node and $\{S_1, S_2, \dots\}$ its children.

The main point made in the paper so far is as follows : we proposed a technique for building S_p so that it explicitely tries to approach all of children models with respect to expected log-likelihood of data to be classified, i.e. it ensures that, for any child $log\ p(D|S_p) \approx log\ p(D|S_c)$. Computation savings come from that $log\ p(D|S_p)$ is the only likelihood that needs to be computed in the classification phase. This seems to us better founded than alternative approaches for fast processing of numerous speaker models, such as anchor models, where Euclidian distances are computed between likelihood vectors.

The further point we make here is that the resulting approximation error may not only be minimized, but also taken into account in order to search the tree more finely, in the classification phase, yet at approximately the same computational cost. Rather than replacing, for all children k, $log\ p(D|S_k)$ by $log\ p(D|S_p)$, the likelihood associated to each child node may be approximated as:

$$\underset{\text{child log-likelihood}}{log\ \tilde{p}(D|S_k)} \approx \underset{\text{parent log-likelihood}}{log\ p(D|S_p)} + \underset{\text{independent of data to classify}}{KL(S_p||S_k)}, k = 1, 2, \dots \quad (17)$$

The main point here is that $KL(S_p||S_k)$ can be pre-computed and is independent of the data to be classified. We advocate the use of the unscented transform (Julier, 1996) for the practical computation of $KL(S_p||S_k)$, as it is more accurate than KL_m used above. This approximation between Gaussian mixtures does not only consider the closest Gaussian, but summarizes each of them by concise statistics, leading to an overall light yet accurate computation. As a side remark, the properties of the unscented transform precluded its use in the model grouping phase.

Because likelihood approximations that are now individual, per child, this second point opens new possibilities for exploring the tree of models, for instance:

(1) searching exhaustively the set of children, by using $log\ \tilde{p}(D|S_k)$, or,
(2) by pre-computing the maximum and minimum error between a parent node and its children :

$$Min_{ERR} = \min_k KL(S_p||S_k), \quad (18)$$

the corresponding cluster of speakers is characterised as having, with high probability, its log-likelihood within $[log\ p(D|S_p) + Min_{ERR}, log\ p(D|S_p) + Max_{ERR}]$, leading to again several possible search schemes.

Exhaustive search	Recognition accuracy		
Query duration (sec)	5	10	15
ML	100%		
KL_m	75%	82.5%	85%

Table 1
Comparing performance of querying exhaustively the collection of speakers, based on maximum likelihood classification (ML line) or computation of KL_m between query and each candidate model (KL_m line). This is examined for 5,10 and 15 seconds queries.

4 Experimental results

All experiments reported below are applied to RealAudio streaming radio broadcast data, in French language. The 13 first MFCC features vectors and their temporal derivates are used. Temporal segmentation of the stream into segments is carried out with the BIC criterion (a classical approximation (Schwarz, 1978) to Bayesian hypothesis testing) over a 4 second sliding window. Individual speaker models are learned using Bayesian adaptation (Bimbot et al., 2004). The stream contains ordinary news programmes, including occasional short jingles than can quite reliably be removed, thanks to their acoustic properties in MFCC space, leaving essentially clean speech sections.

First experiments involve 20 speakers. Accuracy at query phase is evaluated as follows : 40 samples from the 20 speakers are provided for classification (2 per speaker).

We first report an experiment conducting exhaustive search, where query-to-model fitting is conducted by either using definition 15 or maximum likelihood (see table1). While maximum likelihood performs perfectly, KL_m far is less effective (through much faster)

4.1 Results for dendrogram-based hierarchical clustering

Two alternative criteria are compared for measuring similarity between speakers :

- the cross-likelihood, which requires resorting to the feature vectors, which is undesirable from computational cost viewpoint, but should be reliable,
- definition (15), a symmetric version of KL_m.

The trees obtained in these cases are shown in fig. 4.1. It appears that the tree build from KL_m similarity is well-balanced, actually better than the one

Dendrogram-based hierarchy	ML		KL	
Query duration	5	10	5	10
26 Gaussians	92.5%	95%	47.5%	40%
16 Gaussians	95%	95%	50%	45%

Table 2
Recognition accuracy in the case speakers are organized in the tree obtained at fig.4.1b (after cut). Two mixture complexities are considered.

based on cross-likelihood.

When exploring the tree root-to-leaf, query-to-model comparison is evaluated in two cases : (i) likelihood of the query data, given the model, or (ii) symmetric KL_m as in 15. In both cases, the tree was built using definition 15. Results are compared for 2 query lengths (5 and 10 seconds). Results are presened in tab.2. As in the previous experiment, the use of KL_m for querying, instead of ML, implies severe degradation. However, using ML, accuracy remains very satisfactory, and the approach remains beneficial in the sense that : (i) exploration is done through the tree rather than exhaustively, (ii) the tree is built using KL_m, thus quite fast.

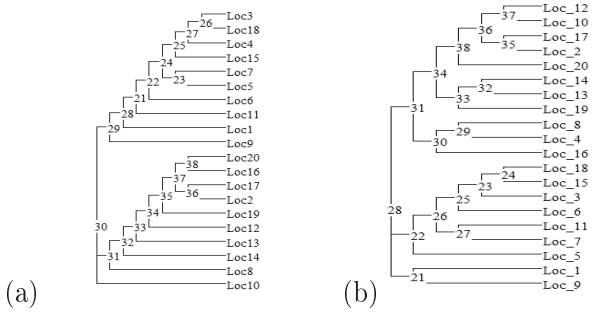


Fig. 2. (a) Binary tree generated using the cross-likelihood scoring matrix between 20 speaker Gaussian mixtures. (b) Binary tree generated using the symmetric KL_m between pairs of speaker models with an incremental perspective.

4.2 Results for a hierarchy build using iterative grouping of mixture models

Table 3 shows recognition accuracy obtained in the same conditions as previous, but the tree is built using the iterative scheme (Algorithm 2) rather than the dendrogram-based approach. The quality of the results are very similar as in the previous approach, which is encouraging, since this iterative approach is far more flexible than the dendrogram-based approach.

Iterative grouping hierarchy	ML		KL	
Query duration	5	10	5	10
26 Gaussians	90%	92.5%	45%	55%
16 Gaussians	92.5%	90%	57.5%	60%

Table 3
Recognition accuracy in the case speakers are organised in the tree obtained by Algorithm 2. Two mixture complexities are considered.

5 Conclusion

In this paper, we addressed the problem of scaling up speaker recognition to a large number of speakers, by organizing the set of speaker models into a search tree. The child-to-parent similarity may be measured and optimized iteratively, using an approximation of KL-divergence, that leads to a low-cost, tractable form. We define and evaluate two ways (dendrogram and iterative grouping) in which this similarity can be exploited, leading to results that loose little reliability with respect to exhaustive search, and offer promising perspectives for speed up. The iterative model grouping procedure is particularly interesting, as is very flexible for incremental processing of the data. There remain to generalise the proposal to more than two levels. Also, the estimated KL divergence between parents and children, which is computed anyway, could provide richer knowledge of the likelihood of data, given children, than is currently done by simply considering the likelihood of data, given the parent.

Acknowledgements

The authors are grateful to the French foreign office for funding this work, as part of the French-Moroccan research network on multimedia (Réseau STIC RTIM).

References

Ben, M., Gravier, G., Bimbot, F., 2004. Enhancing the robustness of bayesian methods for text-independent automatic speaker verification. In: Odyssey'04 Speaker and Language Recognition Workshop. pp. 34–39.

Berrani, S., Amsaleg, L., Gros, P., Nov. 2003. Robust content-based image searches for copyright protection. In: Proc. of ACM workshop on Multimedia databases. New Orleans, USA, pp. 70–77.

Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-Garcia, J., Petrovska-Delacretaz, D., Reynolds, D. A., 4 2004. A tutorial on text-independent speaker verification. EURASIP Journal on Applied Signal Processing (4), 430–451.

Bishop, C., 1995. Neural networks for Pattern Recognition. Oxford University Press.

Chen, M., Shao, M., Ibrahim, J., 2005. Monte Carlo Methods in Bayesian Computation. Springer.

Goldberger, J., Roweis, S., 2004. Hierarchical clustering of a mixture model. In: Proc. of Neural Information Processing Systems (NIPS'2004). pp. 505–512.

Julier, S., Nov. 1996. A general method for approximating a non linear transformation of probability distributions. Tech. rep., Oxford university, Dpt of Engineering Science.

Mami, Y., Charlet, D., Septembre 2002. Speaker identification by location in an optimal space of anchor models. In: International Conferences on Spoken Language Processing (ICSLP '02). Denver, Colorado, USA, pp. 1333–1336.

Nishida, M., Ariki, Y., 1998. Real time speaker indexing based on subspace method - application to tv news articles and debate. In: International Conference on Spoken Language Processing (ICSLP'1998). Sydney, Australia, pp. 1347–1350.

Schwarz, G., 1978. Estimation the dimension of a model. Annals of statistics 6, 461–464.

Sturim, D., Reynolds, D., Singer, D., Campbell, E., 2001. Speaker indexing in large audio databases using anchor models. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01). Salt Lake City, Utah, pp. 429–432.

Upendra, V., Navratil, J., Ramaswamy, G. N., Maes, S., May 2001. Very large population text-independant speaker identification using transformation enhanced multi-grained models. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01). Salt Lake City, Utah, pp. 461–464.

Zezula, P., Amato, G., Dohnal, V., Batko, M., 2006. Similarity Search - The Metric Space Approach. Advances in Database Systems, Vol. 32, 2006, XVIII,. Springer.

Zhou, B., Hansen, J., 2002. Improved structural maximum likelihood eigenspace mapping for rapid speaker adaptation. In: International Conference on Spoken Language Processing (ICSLP'2002). Denver, Colorado, pp. 554–564.

Bibliographie

Akbarinia, R., Martins, V., Pacitti, E. & Valduriez, P. (2006), *Global Data Management*, first edn, IOS Press, chapter Design and Implementation of Atlas P2P Architecture.

Ashbrook, D. & Starner, T. (2002), 'Using gps to learn significant locations and predict movement across multiple users', *Personal and Ubiquitous Computing* **7**(5), 275–286.

Attias, H. (1999), A variational Bayesian framework for graphical models, *in* 'Neural Information Processing Systems (NIPS) Conference', MIT Press, Denver, USA.

Berrani, S.-A., Amsaleg, L. & Gros, P. (2002), 'Recherche par similaritï¿½s dans les bases de donnï¿½es multidimensionnelles : panorama des techniques d'indexation', *Ingï¿½nierie des systï¿½mes d'information* **7**(56), 9–44.

Berrani, S., Amsaleg, L. & Gros, P. (2003), Robust content-based image searches for copyright protection, *in* 'Proc. of ACM workshop on Multimedia databases', New Orleans, USA, pp. 70–77.

Biernacki, C. (1998), Choix de modï¿½les en classification, PhD thesis, Universitï¿½ de Technologie de Compiï¿½gne.

Biernacki, C., Celeux, G. & Govaert, G. (2000), 'Assessing a mixture model for clustering with the integrated completed likelihood', *IEEE Trans. on Pattern Analysis and Machine Intelligence* **22**, 719–725.

Biernacki, C., Celeux, G. & Govaert, G. (2003), 'Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models', *Computational Statistics and Data Analysis* **41**, 561–575.

Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-Garcia, J., Petrovska-Delacretaz, D. & Reynolds, D. A. (2004), 'A tutorial on text-independent speaker verification', *EURASIP Journal on Applied Signal Processing* (4), 430–451.

Bishop, C. (2006), *Pattern recognition and machine learning*, Information science and statistics, Springer.

Bishop, C. & Svensen, M. (2004), Robust Bayesian mixture modelling, *in* 'Proceedings Twelfth European Symposium on Artificial Neural Networks', Bruges, Belgium, pp. 69–74.

Boyd, S., Ghosh, A., Prabhakar, B. & Shah, D. (2006), 'Randomized gossip algorithms', *IEEE Trans. on Information theory* .

Chapelle, O., Schï¿½lkopf, B. & Zien, A. (2006), *Semi-Supervised Learning*, MIT Press.

Cozman, F., Cirelo, M., Huang, T., Cohen, I. & Sebe, N. (2004), 'Semisupervised learning of classifiers : theory, algorithms and their applications to human-computer interaction', *IEEE Trans. on Pattern Analysis and Machine Intelligence* **26**(12), 1553–1567.

Donati, R. & Le Cadre, J.-P. (2006), 'Detection, target motion analysis and track association with a sensor network', *Journal of Information Fusion* **7**(3), 285–303.

Eugster, P., Guerraoui, R., Kermarrec, A.-M. & Massoulïï¿½, L. (2003), 'From epidemics to distributed computing', *IEEE Computer* **37**(5).

Fablet, R., Bouthemy, P. & Perez, P. (2001), 'Non parametric motion characterization using causal probabilistic models for video indexing and retrieval', *IEEE Trans. on Image Processing* **11**(4), 393–407.

Ferecatu, M., Boujemaa, N. & Crucianu, M. (2005), Hybrid visual and conceptual image representation in an active relevance feedback context, *in* 'Proceedings of the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval', Singapour, pp. 6–12.

Gargi, U., Deng, Y. & Tretter, D. R. (2002), Managing and searching personal photo collections, Technical Report HPL-2002-67, HP Laboratories, Palo Alto.

Giannopoulos, E., Streit, R. & Swaszek, P. (1997), Multi-target track segment bearing-only association and ranging, *in* '31rst Asilomar Conference on Signals, Systems and Computers', Pacific Grove.

Goldberger, J. & Aronowitz, H. (2005), A distance measure between gmms based on the unscented transform and its application to speaker recognition, *in* 'Proc. of Interspeech'2005 conference', Lisbon, Portugal.

Graham, A., Garcia-Molina, H., Paepcke, A. & Winograd, T. (2002), Time as essence for photo browsing through personal digital libraries, *in* 'ACM Joint Conference on Digital Libraries JCDL', pp. 326–335.

Hammoud, R. & Mohr, R. (2000), Gaussian mixture densities for video object recognition, *in* 'Proc. on Int. Conf. on Pattern Recognition (ICPR'2000)', Barcelona, Spain, pp. 71–75.

Hayek, R., Raschia, G., Valduriez, P. & Mouaddib, N. (2007), Design of peersum : A summary service for p2p applications., *in* C. Cérin & K.-C. Li, eds, 'GPC', Vol. 4459 of *Lecture Notes in Computer Science*, Springer, pp. 13–26.

Hsu, C.-W. & Lin, C.-J. (2002), 'A comparison of methods for multiclass support vector machines', *IEEE Transactions on Neural Networks* **13**(2), 415–425.

Kempe, D., Dobra, A. & Gehrke, J. (2003), Gossip-based computation of aggregate information, *in* 'IEEE symp. on foundations of computer science', Cambridge, MA, USA.

Kennedy, L., Naaman, M., Ahern, S., Nair, R. & Rattenbury, T. (2007), How flickr helps us make sense of the world : Context and content in community-contributed media collections, *in* 'ACM International Conference on Multimedia (ACM MM 2007),', Augsburg, Germany.

Kokaram, A., Rea, N., Dahyot, R., Tekalp, M., Bouthemy, P., Gros, P. & Sezan, I. (2006), 'Browsing sports video (trends in sports-related indexing and retrieval work)', *IEEE Signal Processing Magazine* **23**(2), 47–58.

Kowalczyk, W. & Vlassis, N. (2005), Newscast EM, *in* M. Press, ed., 'Proc. of Neural Information Processing Systems (NIPS) 17'.

Lasserre, J. A., Bishop, C. M. & Minka, T. P. (2006), Principled hybrids of generative and discriminative models, *in* '2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA', pp. 87–94.

Mami, Y. & Charlet, D. (2002), Speaker identification by location in an optimal space of anchor models, *in* 'International Conferences on Spoken Language Processing (ICSLP '02)', Denver, Colorado, USA, pp. 1333–1336.

Manjarrez, J., Martinez, J. & Valduriez, P. (2007), A data allocation method for efficient content-based retrieval in parallel multimedia databases, *in* 'International Symposium on Parallel and Distributed Processing and Applications', Niagara Falls, Canada.

McLachlan, G. & Peel, D. (2000), *Finite mixture models*, Wiley.

Milojicic, D., Kalogeraki, V., Lukose, R., Nagaraja, L., Pruyne, J., Richard, B., Rollins, S. & Xu, Z. (2002), Peer-to-peer computing, Technical Report 2002-57, Hewlett-Packard.

Neal, R. & Hinton, G. (1998), *Learning in graphical models*, Dordrecht : Kluwer academic publishers, chapter A view of the EM algorithm that justifies incremental, sparse and other variants, pp. 355–368.

Nilsback, M.-E. & Zisserman, A. (2006), A visual vocabulary for flower classification, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition'. To appear.

Nowak, R. (2003), 'Distributed EM algorithms for density estimation and clustering in sensor networks', *IEEE Trans. on Signal Processing* **51**(8).

Philbin, J., Chum, O., Isard, M., Sivic, J. & Zisserman, A. (2007), Object retrieval with large vocabularies and fast spatial matching, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition'.

Platt, J. C. & M. Czerwinski, B. A. F. (2002), PhotoTOC : Automatic clustering for browsing personal photographs, Technical Report MSR-TR-2002-17, Microsoft Research.

Ponce, J., Hebert, M., Schmid, C. & Zisserman, A. (2006), *Towards category-level object recognition*, Springer.

Poullot, S., Buisson, O. & Crucianu, M. (2007), Z-grid-based probabilistic retrieval for scaling up content-based copy detection, *in* 'ACM International Conference on Image and Video Retrieval', Amsterdam.

Ramanan, D., Forsyth, D. & Barnard, K. (2006), 'Building models of animals from video', *IEEE Trans. on Pattern Analysis and Machine Intelligence* **28**(8), 1319–1334.

Reynolds, D. (1995), 'Speaker identification and verification using gaussian speaker models', *Speech communication* **17**, 91–108.

Rodden, K. (2003), How do people manage their digital photographs ?, *in* 'ACM Conference on Human Factors in Computing Systems', Fort Lauderdale, pp. 409 – 416.

Rowley, H., Baluja, S. & Kanade, T. (1998), 'Neural-network based face detection', *IEEE Trans. on Pattern Analysis and Machine Intelligence* **20**(1), 23–38.

Santoro, N. (2006), *Design and Analysis of Distributed Algorithms (Wiley Series on Parallel and Distributed Computing)*, Wiley-Interscience.

Schmid, C. (2004), 'Weakly supervised learning of visual models and its application to content-based retrieval', *Int. Journal of Computer Vision* **1**(56), 7–16.

Sclaroff, S., La Cascia, M., Sethi, S. & Taycher, L. (1999), 'Unifying textual and visual cues for content-based image retrieval on the world wide web', *Computer Vision and Image Understanding* **75**, 86–98.

Sturim, D., Reynolds, D., Singer, D. & Campbell, E. (2001), Speaker indexing in large audio databases using anchor models, *in* 'IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)', Salt Lake City, Utah, pp. 429–432.

Tipping, M. (2002), 'Sparse bayesian learning and the relevance vector machine', *Journal of Machine Learning Research* **1**, 211–244.

Tipping, M. & Bishop, C. (1999), 'Mixtures of probabilistic principal component analysers', *Neural Computation* **11**(2), 443–482.

Tipping, M. E. (2001), 'Sparse bayesian learning and the relevance vector machine.', *Journal of Machine Learning Research* **1**, 211–244.

Vasconcelos, N. & Lippman, A. (1998), Learning mixture hierarchies, *in* 'Neural Information Processing Systems (NIPS) Conference', Denver, Colorado.

Wactlar, H., Kanade, T., Smith, M., Stevens & S.Smith (1996), 'Intelligent access to digital video : the informedia project', *IEEE Computer* **29**(5).

Wellner, P., Flynn, M. & Guillemot, M. (2004), Browsing recordings of multi-party interactions in ambient intelligent environments, *in* 'ACM CHI'2004 Conference (Computer-Human Interaction)', Vienna, Austria, pp. 120–126.

Williams, O., Blake, A. & Cipolla, R. (2005), A sparse probabilistic learning algorithm for real-time tracking, *in* 'IEEE Trans. on Pattern Analysis and Machine Intelligence', Vol. 27, pp. 1292–1304.