
Résumé

Motivation

L'objectif de cette thèse est de contribuer au développement de méthodes de reconnaissance d'objets dans les images, s'appuyant sur des informations de haut niveau, sur la structure de la scène (topologie, organisation spatiale des différents objets dans l'image).

Au cours des dernières années les graphes ont été utilisés pour représenter mathématiquement la connaissance structurelle. L'intérêt de ce type de représentation va en augmentant dans la communauté scientifique grâce à la possibilité d'utiliser cette représentation en combinaison avec des algorithmes de mise en correspondance de graphes (*graph matching* en anglais), car cette représentation permet de représenter les variations et les différences structurelles entre objets différents.

Différentes applications ont été proposées en utilisant ce type de représentation pour réaliser la reconnaissance automatique des images. Dans ce type d'applications, la connaissance sur les variations structurelles possibles des objets qui apparaissent dans les images (personne, visage, image médicale, etc.) est exprimée à travers un modèle que l'on représente comme un graphe (ou atlas, comme on l'appelle dans certains cas). La cartographie, la reconnaissance de caractères, et la reconnaissance de structures cérébrales à partir d'images de résonance magnétique sont des exemples de domaines dans lesquels ce type de représentation a été utilisé et publié. Ce dernier exemple est illustré figure 9.

Le *graphe modèle* est souvent construit en utilisant un nœud pour chacune des régions de l'objet à reconnaître (Ex: dans le cas d'images d'un cerveau humain, il y aura un nœud pour représenter le cervelet, un autre pour chaque ventricule...) et les arêtes pour représenter les relations entre ces régions. On utilise des attributs pour exprimer les propriétés de chaque nœud ou arête, et pour pouvoir les identifier ainsi que les distinguer les uns des autres. Ce graphe est souvent construit de manière supervisée.

Après la construction d'un graphe modèle, et pour pouvoir réaliser la reconnaissance d'images à travers lui, il est nécessaire de construire un graphe à partir de chacune des images à reconnaître. Ces graphes, qui sont dénommés dans la littérature *graphes de données* ou *graphes d'entrée*, sont automatiquement construits par l'ordinateur sans l'assistance de l'utilisateur. Dans ce processus de construction du graphe de données correspondant à une image, une étape de segmentation de l'image précède la construction du graphe et est fondamentale : le graphe de données est formée en utilisant un nœud pour représenter chacun des segments obtenus. Les mêmes attributs que ceux du graphe modèle sont calculés pour les nœuds et les arêtes du graphe de données.

Mise en correspondance de graphes

La reconnaissance des structures de l'image est réalisée en recherchant la meilleure correspondance entre les graphes modèle et de données. Il s'agit d'affecter une étiquette à chacune des régions de l'image à reconnaître, les étiquettes étant définies par les régions du graphe modèle. La figure 10 illustre le principe de la mise en correspondance.

En notant $G_M = (V_M, E_M)$ le graphe modèle et $G_D = (V_D, E_D)$ le graphe de données, le problème consiste à trouver un homomorphisme $h : V_D \rightarrow V_M$ tel que $(u, v) \in E_D$ si et seulement si $(f(u), f(v)) \in E_M$. Ces homomorphismes sont évalués en calculant la similarité entre les attributs des nœuds et des arêtes mis en correspondance. On cherche alors un meilleur homomorphisme au sens d'un certain critère combinant ces similarités.

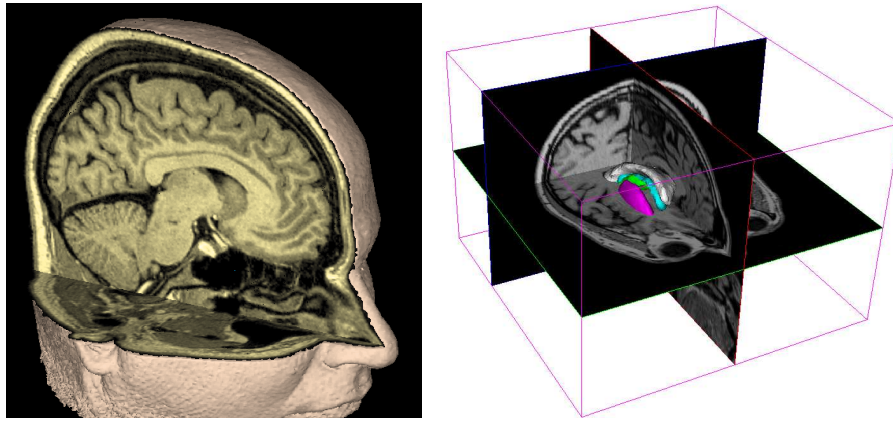


Figure 9: Exemple d'application de la mise en correspondance de graphes appliquée à la reconnaissance d'images. L'objectif est de reconnaître les différentes parties d'un cerveau humain à partir d'une image 3D comme celle de gauche.

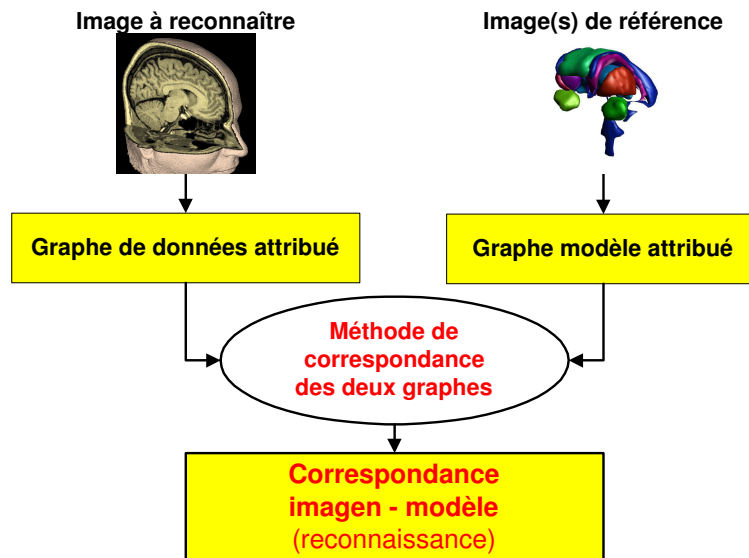


Figure 10: Principe de la mise en correspondance des graphes modèle et de données pour la reconnaissance de structures cérébrales.

Une grande partie de la littérature est consacrée à la recherche d'isomorphismes de graphes, ce qui suppose en particulier que $|V_M| = |V_D|$. On parle alors de correspondance exacte. Cependant, dans beaucoup de problèmes, surtout pour ceux où la segmentation s'effectue automatiquement, la condition d'isomorphisme est trop stricte et le nombre de nœuds du graphe de données est supérieur à celui du modèle ($|V_M| < |V_D|$). On parle alors de *correspondance inexacte de graphes*, et la complexité de ces problèmes est supérieure à celle des précédents. C'est à ce type de problème que nous nous intéressons ici.

La figure 11 illustre les différents types de problèmes de mise en correspondance de graphes. Dans les problèmes de mise en correspondance inexacte, on utilise souvent :

- un nœud nul ou *dummy* représentant la non-reconnaissance (par exemple pour une

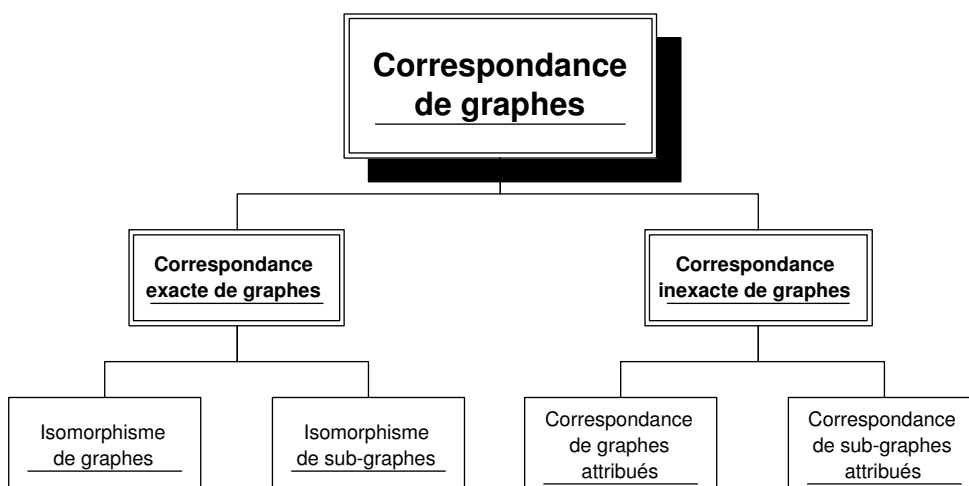


Figure 11: Classement des problèmes de mise en correspondance de graphes en deux classes principales.

région de l'image correspondant à une structure non représentée dans le modèle) ;

- des correspondances multiples entre nœuds : ces types de problèmes sont les plus complexes, car il est possible qu'un segment de l'image à reconnaître fasse en même temps partie de plus d'une région de l'objet à traiter. Bien que ce type de problèmes puisse exister, leur résolution implique plus de combinaisons et la recherche pour trouver la solution optimale peut être de complexité trop élevée pour pouvoir être traitée.

Beaucoup de techniques très différentes ont été développées pour la mise en correspondance de graphes : arbres de décision [Messmer and Bunke, 1999], réseaux de neurones [Riviere et al., 2002], algorithme EM [Cross and Hancock, 1998], relaxation probabiliste [Christmas et al., 1995], heuristiques et métaheuristiques [Pelillo et al., 1999], algorithmes génétiques [Wilson and Hancock, 1997], et programmation évolutive [Singh and Chaudhury, 1997]. Beaucoup de ces techniques restent restreintes aux correspondances exactes de graphes. Cependant, dans beaucoup d'autres exemples, surtout en ce qui concerne les applications aux images réelles, la condition d'isomorphisme est trop forte, cependant les approches pour résoudre les problèmes de correspondances inexacts sont actuellement moins répandues.

Les problèmes considérés dans cette thèse sont des problèmes de mise en correspondance inexacte de graphes, sans ajouter de nœud nul. En outre, dans le but de ne pas augmenter la complexité de problèmes qui sont déjà complexes, il a été décidé de réaliser une sursegmentation de l'image à reconnaître pour éviter la possibilité de correspondances multiples entre graphes. La méthode proposée pour réaliser la mise en correspondance repose sur les algorithmes d'estimation de distributions (EDAs).

Formalisation de la mise en correspondance de graphes comme un problème d'optimisation combinatoire

A cause de la grande complexité du problème de mise en correspondance inexacte, nous nous sommes tournés vers des algorithmes d'optimisation combinatoire sous contrainte. Cette thèse analyse les aspects et mécanismes qui doivent s'appliquer pour pouvoir formuler le

problème de cette manière et ensuite appliquer des algorithmes tels que les Algorithmes d'Estimation de Distributions (EDAs).

Les caractéristiques suivantes doivent être définies :

- **une représentation des individus** ou des solutions, de manière à représenter chacun des points dans l'espace de recherche ;
- **une fonction objectif à optimiser** qui affecte une valeur à chaque solution possible ou individu, pour exprimer la validité et la qualité de l'individu comme solution au problème.

Chaque individu est composé d'un vecteur de valeurs qui peuvent être discrètes ou continues. Les EDAs permettent d'utiliser les deux types d'individus contrairement à d'autres algorithmes. Cette thèse présente la façon de travailler avec les deux types d'individus et de représentations afin de les appliquer dans des problèmes de mise en correspondance de graphes.

Une représentation possible des individus consiste à associer à chaque nœud de G_D un nœud de G_M . Donc, la taille des individus est déterminée dans ce cas par $n = |V_D|$ variables⁴, $X_i \in \mathbf{X}$ $i = 1, \dots, |V_D|$, où chaque variable contient une valeur entre 1 et $|V_M|$. La valeur de chaque variable dans l'individu a la signification suivante : $X_i = k$ avec $1 \leq i \leq |V_D|$ et $1 \leq k \leq |V_M| \Leftrightarrow$ le i -ème nœud de G_D est identifié comme le k -ème nœud du graphe modèle G_M .

Cette thèse propose deux autres représentations d'individus dans le domaine discret, ainsi qu'une représentation pour le domaine continu. En outre, et avec l'intention de montrer la robustesse de la méthode proposée, une contrainte supplémentaire que doivent accomplir les individus pour pouvoir être pris en considération est ajoutée :

$$\forall a_M \in V_M, \exists a_D \in V_D \mid h(a_D) = a_M$$

On impose ainsi que toute structure du modèle soit identifiée dans l'image. Cette restriction augmente ainsi la complexité des problèmes et celle des différents mécanismes présentés dans cette thèse. Elle permet de montrer comment on peut tenir compte des restrictions quand on applique les EDAs.

Enfin, dans le but d'évaluer la qualité de l'individu comme solution possible à un problème, on définit deux fonctions $c_N(a_D, a_M)$ et $c_E(e_D, e_M)$ qui mesurent respectivement la similarité entre deux nœuds $a_D \in V_D$ et $a_M \in V_M$, et entre deux arêtes $e_D \in E_D$ et $e_M \in E_M$. Nous avons défini plusieurs fonctions objectif dépendant de ces fonctions de similarité.

Algorithmes d'Estimation de Distributions

Les Algorithmes d'Estimation de Distributions (EDAs) sont une approche récente dans la famille des algorithmes évolutifs appliqués aux problèmes d'optimisation. Ces algorithmes se caractérisent par un procédé d'évolution d'un ensemble de solutions. À cause de la similitude avec des populations d'êtres vivants, on appelle d'habitude *individu* chacune des solutions, *population* l'ensemble d'individus, et *génération* chacune des populations qui se développent successivement. La différence la plus significative entre les EDAs et les Algorithmes Génétiques (GAs), qui sont parmi les plus connus et utilisés dans ce domaine, réside

⁴Dans le domaine des algorithmes génétiques on considère que les individus sont composés de gènes, mais dans le domaine des EDAs on parle plutôt de variables probabilistes.

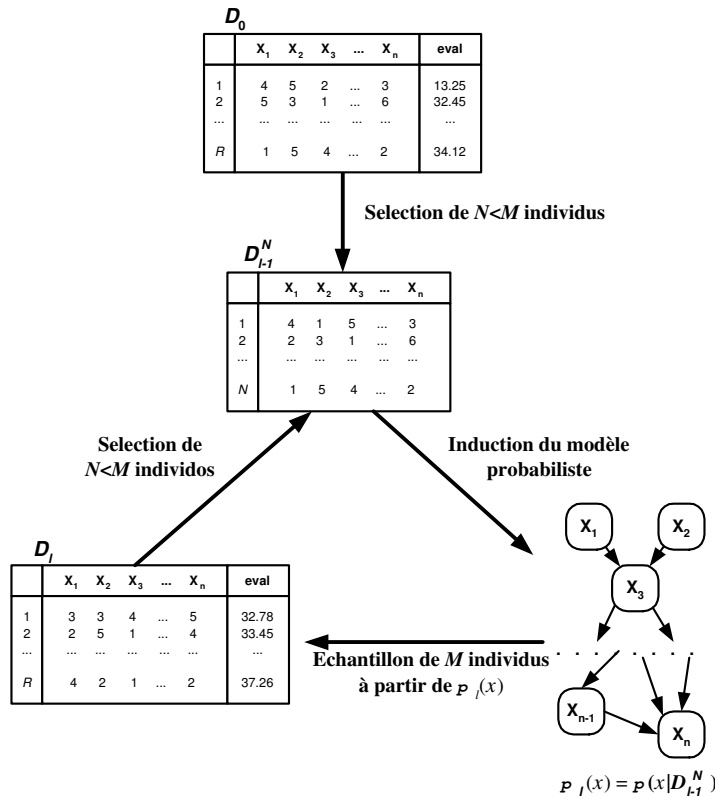


Figure 12: Illustration de l'approche EDA en optimisation.

dans le processus d'évolution d'une génération à la suivante : elle est réalisée dans le cas des GAs par des opérations de mutation et de croisement, et dans les EDAs par des techniques fondées sur la théorie des probabilités et plus particulièrement sur l'apprentissage et la simulation de réseaux bayésiens ou de réseaux gaussiens. Cette idée est illustrée figure 12.

La figure 13 présente le pseudocode générique des EDAs qui suit essentiellement les étapes suivantes :

1. La population initiale D_0 formée par R individus est créée. La génération de ces R individus est souvent réalisée en supposant une distribution uniforme pour chaque variable. Une fois que les individus sont engendrés, ils sont évalués selon la fonction objectif.
2. Pour faire évoluer la i -ème population D_{l-1} vers la suivante D_l , N individus sont sélectionnés dans D_{l-1} ($N < R$) en suivant un certain critère. Nous appelons D_{l-1}^N l'ensemble des N individus sélectionnés dans la génération numérotée $l - 1$.
3. Le modèle probabiliste graphique n -dimensionnel qui représente le mieux les dépendances entre les n variables est déterminé. C'est l'étape d'*apprentissage*, qui est cruciale dans les EDAs à cause de l'importance de la prise en compte de toutes les dépendances entre les variables pour assurer une évolution satisfaisante vers des individus meilleurs.
4. La nouvelle population D_l est constituée avec R nouveaux individus obtenus après avoir simulé la distribution de probabilité apprise dans l'étape précédente. On utilise souvent une approche élitiste, et ainsi le meilleur individu de la population D_{l-1}^N est

EDA

$D_0 \leftarrow$ Engendrer R individus (la population initiale D_0) au hasard

Répéter pour $l = 1, 2, \dots$ jusqu'à satisfaire un critère d'arrêt

$D_{l-1}^N \leftarrow S$ $N < R$ individus de D_{l-1} en suivant
une méthode de sélection déterminée

$\rho_l(\mathbf{x}) = \rho(\mathbf{x} | D_{l-1}^N) \leftarrow$ Estimer la distribution de probabilité
qu'un individu se trouve entre les individus sélectionnés

$D_l \leftarrow$ Echantillonner R individus, la nouvelle population, à partir de $\rho_l(\mathbf{x})$

Figure 13: Pseudocode générique des EDAs.

gardé dans la nouvelle population D_l . De cette manière, dans chaque génération un total de $R - 1$ nouveaux individus sont créés au lieu de R .

Les étapes 2, 3 et 4 sont répétées jusqu'à satisfaire une condition d'arrêt. Des exemples de critères d'arrêt sont : arriver à un nombre de génération maximal, atteindre un nombre maximal d'individus analysés, uniformité dans la population, ou le fait de ne pas obtenir un individu avec une valeur de fonction objectif meilleure après un certain nombre de générations.

Un grand nombre d'algorithmes qui font partie des EDAs a été proposé dans la littérature. On peut les classer en trois grands groupes selon la complexité du type de dépendances entre variables considéré :

- **Sans dépendance entre variables** : la structure du réseau bayésien (ou gaussien si nous travaillons dans le domaine continu) est fixe et ne contient pas d'arcs. En d'autres termes, cela signifie que toutes les variables de cet individu sont considérées indépendantes entre elles.

Comme exemple d'algorithmes du domaine discret qui appartiennent à ce groupe nous pouvons mentionner UMDA (Univariate Marginal Distribution Algorithm) [Mühlenbein, 1998] où l'estimation de la distribution de probabilité est réalisée de la manière suivante :

$$p(\mathbf{x}) = \prod_{i=1}^n p(x_i)$$

Un autre exemple est l'algorithme connu comme UMDA_c (Univariate Marginal Distribution Algorithm - continuous) [Larrañaga et al., 2000], qui est l'équivalent de l'UMDA dans le domaine continu.

- **Dépendances entre paires de variables** : dans la structure du réseau bayésien ou gaussien chacune des variables peut avoir au plus un parent. Cela nécessite donc une étape d'estimation de la meilleure structure.

Un exemple d'EDAs pour le domaine discret appartenant à ce groupe est MIMIC (Mutuel Information Maximization for Input Clustering) [de Bonet et al., 1997] qui propose de réaliser la factorisation de probabilité suivante :

$$p(\mathbf{x}) = p(x_{i_1} | x_{i_2}) \cdot p(x_{i_2} | x_{i_3}) \cdots p(x_{i_{n-1}} | x_{i_n}) \cdot p(x_{i_n})$$

MIMIC recherche la permutation $\pi = (i_1, i_2, \dots, i_n)$ qui minimise la divergence de Kullback-Leibler entre la distribution estimée $\hat{p}_\pi(\mathbf{x})$ et la distribution réelle $p(\mathbf{x})$.

De nouveau, il existe une version continue de MIMIC nommée MIMIC_c (Mutuel Information Maximization for Input Clustering - continuous) [Larrañaga et al., 2000].

- **Dépendances multiples entre variables :** cette fois toutes les probabilités conditionnelles peuvent intervenir, et la structure du réseau bayésien ou gaussien est quelconque. Cette caractéristique demande une recherche exhaustive de la meilleure structure probabiliste graphique parmi toutes celles qui sont possibles, et donc ces algorithmes sont plus chers en termes de temps d'exécution que ceux des groupes précédents. Leur avantage est qu'ils sont capables d'apprendre une structure probabiliste graphique qui considère les relations entre les différentes variables des individus plus fidèlement.

Citons comme exemple dans le domaine discret l'algorithme EBNA (Estimation of Bayesian Networks Algorithm) [Etzeberria and Larrañaga, 1999], et EGNA (Estimation of Gaussian Networks Algorithm) [Larrañaga et al., 2000, Larrañaga and Lozano, 2001] dans le domaine continu.

L'algorithme EBNA utilise un *score* BIC (Bayesian Information Criterion) [Schwarz, 1978] qui mesure l'aptitude d'une structure à représenter les dépendances entre les individus. Le réseau bayésien qui maximise cette mesure est estimé par la méthode de l'Algorithme B [Buntine, 1991]. La factorisation de la probabilité s'écrit alors de la façon suivante :

$$p(\mathbf{x}) = \prod_{i=1}^n p(x_i | \mathbf{pa}(x_i))$$

où $\mathbf{pa}(x_i)$ est l'ensemble des parents de la variable x_i dans le réseau bayésien.

Les algorithmes et approches d'EDAs qui sont proposés dans la littérature sont nombreux, bien que pour le moment il y n'ait que peu d'articles qui en montrent tout le potentiel en comparaison avec d'autres paradigmes plus connus comme par exemple les GAs et les Stratégies Evolutives (ES). Cette thèse vise à combler ce manque dans le cas de la mise en correspondance de graphes pour la reconnaissance d'objets dans les images. Un autre des aspects innovants de cette thèse est précisément le fait d'appliquer pour la première fois les EDAs aux problèmes de mise en correspondance de graphes, tout en tenant compte des deux domaines discret et continu.

Dans ce cas, les distributions estimées dans les EDAs représentent les dépendances entre les différentes correspondances possibles entre nœuds de G_D par rapport aux nœuds de G_M .

EDAs parallèles

Une des principales limites des EDAs pour les problèmes traités ici est le temps de calcul important qu'ils nécessitent pour aboutir à la solution. Nous avons donc étudié les possibilités de parallélisation de ces algorithmes afin de réduire ces temps d'exécution.

Nous avons analysé différents EDAs en fonction des besoins de calcul pour chacune de leurs étapes. Cette analyse a été réalisée avec l'outil GNU *gprog*. Ces études soulignent que les algorithmes les plus chers en calcul sont ceux du troisième groupe, et cet effet est précisément dû à l'étape d'apprentissage. Dans le cas de la reconnaissance de structures

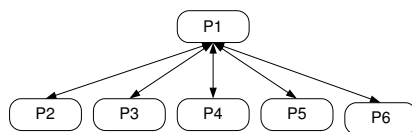


Figure 14: Schéma d'exécution maître-esclave, où un processus joue le rôle de *maître* de tâches et les autres réalisent des parties d'un travail commun.

cérébrales, cet apprentissage avec EBNA prend 85,7% du temps d'exécution total. Nous avons décidé de paralléliser l'algorithme EBNA (troisième groupe et domaine discret).

L'apprentissage dans EBNA repose sur la mesure BIC, qui demande presque tout le temps de calcul de cet apprentissage. La mesure $BIC(S, D)$, où S est la structure et D les données à partir desquelles est réalisé l'apprentissage, peut être divisée en composantes $BIC(i, S, D)$ qui expriment la mesure locale BIC pour une variable X_i :

$$BIC(S, D) = \sum_{i=1}^n BIC(i, S, D) = \sum_{i=1}^n \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log \frac{N_{ijk}}{N_{ij}} - \frac{1}{2}(r_i - 1)q_i$$

Pour paralléliser le programme on utilise le modèle d'exécution maître-esclave illustré figure 14, dans lequel chacun des esclaves calcule les termes $BIC(i, S, D)$ correspondant aux différentes variables X_i $i = 1, \dots, n$.

Pour essayer des techniques distinctes de parallélisme, cette thèse utilise des bibliothèques pthreads et MPI qui permettent de tester l'intérêt d'utiliser la *mémoire partagée* et le *passage des messages* respectivement, avec des configurations différentes des ordinateurs.

Les essais réalisés montrent qu'avec des threads et avec du MPI nous pouvons obtenir des réductions significatives du temps de calcul, allant jusqu'à 60%.

Résultats expérimentaux

Nous avons traité différents exemples d'application des EDAs à des problèmes de correspondance inexacte de graphes. Des exemples synthétiques obtenus par simulations permettent d'illustrer différentes caractéristiques des EDAs et des techniques proposées :

- Comparaison entre EDAs, algorithmes génétiques et stratégies évolutives pour des problèmes de complexités très différentes.
- Analyse de quatre méthodes adaptées aux EDAs pour satisfaire les contraintes imposées aux problèmes de correspondance de graphes.
- Etude de l'évolution des structures graphiques probabilistes de génération en génération pour les EDAs discrets ou continus.
- Application et mesure de rendement des techniques de parallélisme proposées.

Des applications réelles ont ensuite été traitées, portant sur la reconnaissance de structures cérébrales à partir d'images 3D de résonance magnétique, et de structures faciales (il s'agit de trouver le nez, la bouche, etc.). Les résultats obtenus pour ce dernier problème pour les algorithmes ssGA (algorithme génétique), UMDA et EBNA sont illustrés figure 15.

Dans tous les exemples réels traités, les erreurs de reconnaissance que peuvent commettre les techniques de correspondance entre graphes se divisent en trois types, illustrés figure 16.

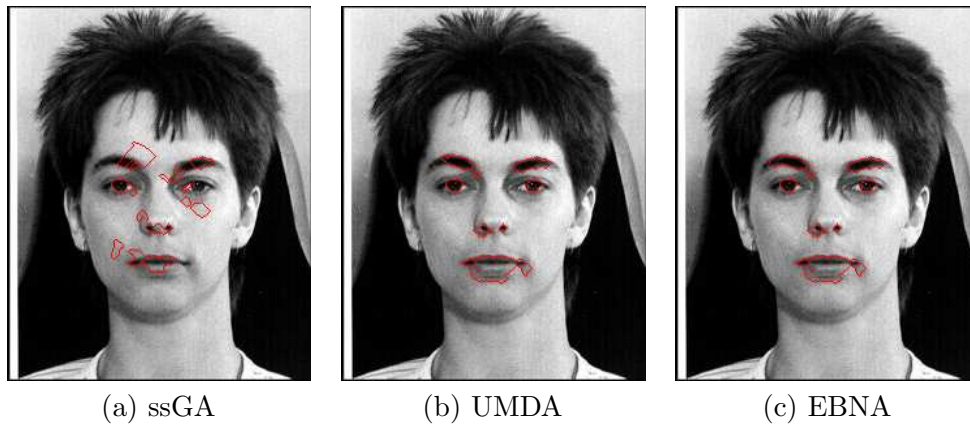


Figure 15: Exemple d'un problème réel de reconnaissance de structures faciales. Les résultats sont obtenus en utilisant l'algorithme génétique SSGA et les EDAs UMDA et EBNA. Ces résultats montrent que la reconnaissance est meilleure dans le cas des EDAs.

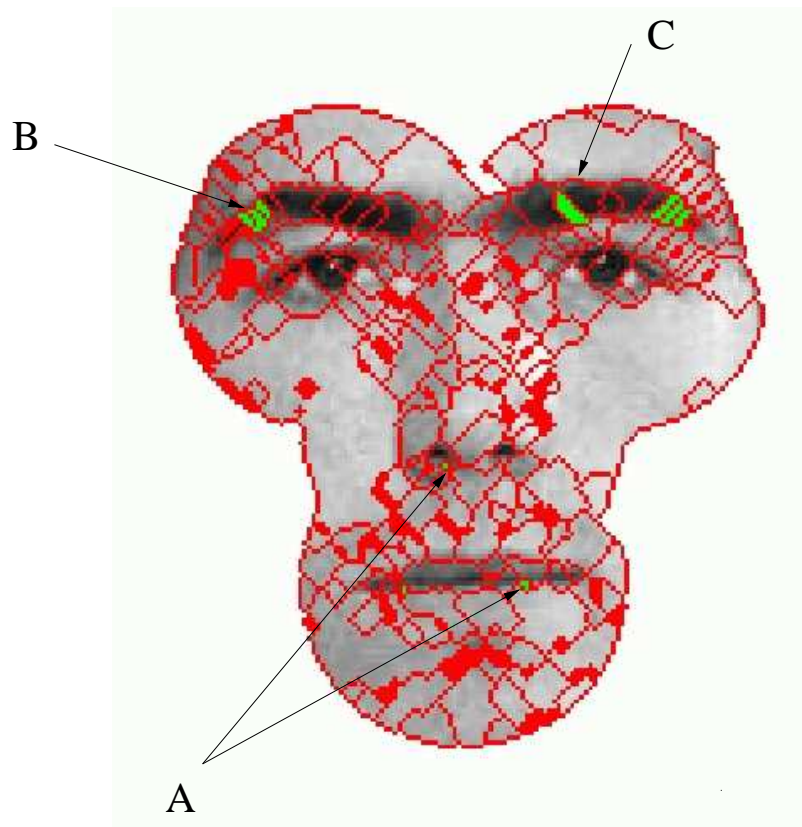


Figure 16: Exemple de quelques erreurs typiques de reconnaissance. A: segments de taille trop petite pour pouvoir être identifiés, même visuellement. B: segments situés à proximité des limites de deux régions ou plus, trop ambigus pour pouvoir être classés. C: vraies erreurs de reconnaissance.

Conclusions et perspectives

Cette thèse formalise les problèmes de mise en correspondance de graphes comme des problèmes d'optimisation combinatoire sous contraintes. Un des apports les plus innovants de cette thèse consiste à utiliser pour la première fois les EDAs dans ce type de problèmes.

Quant à la comparaison des EDAs avec des GAs, la conclusion des essais réalisés est la suivante : pour des problèmes qui ne sont pas très complexes les GAs donnent des résultats similaires à ceux des EDAs et nécessitent moins de temps d'exécution ; cependant, pour des problèmes complexes les EDAs donnent toujours de meilleurs résultats que les GAs, qui sont plus susceptibles de tomber dans des extrema locaux. En outre, les EDAs du domaine continu se comportent généralement mieux que ceux du domaine discret quant à la qualité de la solution obtenue, bien qu'avec un coût plus grand en termes de temps d'exécution.

Différents types de fonctions objectif ont également été proposés, fondés sur des paradigmes différents comme la théorie des ensembles flous et la théorie des probabilités. Les résultats montrent aussi l'importance de choisir une bonne façon de générer le graphe modèle et de définir les attributs.

Finalement, les résultats expérimentaux montrent que la parallélisation des EDAs contribue de façon satisfaisante à la réduction de temps de calcul.

Les perspectives de ce travail sont les suivantes :

Modélisation : considérer les différences d'importance entre les régions, engendrer des modèles à partir de plus d'une image.

Fonctions objectif : réaliser une comparaison de rendement entre elles, tester d'autres représentations d'individus et de fonctions objectif.

Vérifier la validité de la méthode sur des séquences d'images et exploiter la continuité d'une image à la suivante pour améliorer la reconnaissance.

Amélioration des EDAs : autres techniques de génération de populations initiales, essais avec d'autres modèles graphiques probabilistes.

Parallélisation des EDAs : paralléliser d'autres EDAs, combinaison entre algorithmes en parallèle, appliquer d'autres techniques de parallélisme.

Publications des travaux réalisés pendant la thèse

Cette thèse a donné lieu à plusieurs publications dans des revues et congrès internationaux.

2002

- Bengoetxea, E., Larrañaga, P., Bloch, I., Perchant, A., and Boeres, C. (2002). Learning and simulation of Bayesian networks applied to inexact graph matching. *Pattern Recognition* 35(12):2867-2880.
- Cesar R., Bengoetxea E., Bloch I. Inexact graph matching using stochastic optimization techniques for facial feature recognition. In *International Conference on Pattern Recognition (ICPR 2002)*. Quebec, Canada.
- Cesar R., Bengoetxea E., Bloch I., Larrañaga, P. Inexact graph matching for facial feature recognition. *International Journal of Imaging Systems and Technology* (submitted).

-
- Mendiburu A., Bengoetxea, E., Miguel J. Paralelización de algoritmos de estimación de distribuciones. In XIII Jornadas de Paralelismo, pages 37-41. ISBN: 64-8409-159-7.

2001

- Bengoetxea, E., Larrañaga, P., Bloch, I., and Perchant, A. (2001). Estimation of Distribution Algorithms: A New Evolutionary Computation Approach For Graph Matching Problems. Lecture notes in Computer Science 2134. Third International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR-2001). M. Figueiredo, J. Zerubia and A. K. Jain (eds.), pages 454-468, Sophia Antipolis, France.
- Bengoetxea, E., Larrañaga, P., Bloch, I., and Perchant, A. (2001). Image Recognition with Graph Matching Using Estimation of Distribution Algorithms. Proceedings of the Medical Image Understanding and Analysis (MIUA 2001), 89-92, Birmingham, UK.
- Bengoetxea, E., Larrañaga, P., Bloch, I., Perchant, P. (2001). Solving graph matching with EDAs using a permutation-based representation. Estimation of Distribution Algorithms. A new tool for Evolutionary Computation, Larrañaga, P. and Lozano, J.A. (eds.). Kluwer Academic Publishers.
- Larrañaga, P., Bengoetxea, E., Lozano, J.A., Robles, V., Mendiburu, A., and de Miguel, P. (2001). Searching for the Best Permutation with estimation of Distribution Algorithms. Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI 2001). Workshop on Stochastic Search Algorithms, pages 7-14, Seattle, USA.
- Bengoetxea, E., Miquélez, T., Lozano, J. A., and Larrañaga, P. (2001). Experimental results in function optimization with EDAs in continuous domain. Estimation of Distribution Algorithms. A new tool for Evolutionary Computation, Larrañaga, P. and Lozano, J.A. (eds.). Kluwer Academic Publishers.
- Larrañaga, P., Lozano, J. A., and Bengoetxea, E. (2001). Estimation of Distribution Algorithms based on multivariate normal and Gaussian networks. Technical Report KZZA-IK-1-01 of the Department of Computer Science and Artificial Intelligence, University of the Basque Country, Spain.
- Miquélez, T., Bengoetxea, E., Morlán, I., and Larrañaga, P. (2001). Obtention of filters for Image Restoration Using Estimation of Distribution Algorithms. CAEPIA 2001, Spanish Society for the Artificial Intelligence. (in Spanish).

2000

- Bengoetxea, E., Larrañaga, P., Bloch, I., Perchant, A., and Boeres, C. (2000). Inexact graph matching using learning and simulation of Bayesian networks. An empirical comparison between different approaches with synthetic data. Proceedings of CaNew workshop, ECAI 2000 Conference, ECCAI. Berlin, Germany.

