Fouille d'images IRMf: algorithme CURE

Jerzy Korczak, Aurélie Bertaux

Fouille d'images IRMf : algorithme CURE

Jerzy Korczak, Aurélie Bertaux

LSIIT, Bd Sébastien Brant, 67412 Illkirch cedex France

<korczak, bertaux>@lsiit.u-strasbg.fr

Résumé. Dans cet article, nous présentons la fouille d'images IRMf à travers l'utilisation d'un algorithme hiérarchique ascendant de classification non supervisée : CURE. Nous avons adapté cet algorithme à nos contraintes pour nous permettre de traiter des volumes de données importants en temps quasi réel. Ces extensions portent sur le tirage aléatoire des signaux, leur partitionnement, leur échantillonnage, ainsi que la représentativité des clusters.

Nous exposerons les divers éléments de tests réalisés sur des données IRMf synthétiques, et nous critiquerons les resultats obtenus par CURE en dévoilant ses forces et faiblesses, en comparaison avec d'autres algorithmes implémentés préalablement et testés sur les mêmes données.

1 Introduction à la fouille d'images IRMf

L'Imagerie par Résonance Magnétique (IRM) produit des coupes virtuelles afin de visualiser l'anatomie du cerveau. Ces images d'une précision extrême permettent à des radiologues de détecter et localiser d'éventuelles lésions cérébrales chez leurs patients. L'IRM s'est spécialisée dans plusieurs branches dont récemment l'IRM fonctionnelle (IRMf), qui facilite la distinction des différentes parties du cerveau qui s'activent selon la "fonction" qui leur est associée. C'est sur ce principe que se base notamment l'imagerie neurofonctionnelle IRMf car le cerveau n'est pas une masse homogène mais elle est divisée en régions plus ou moins spécialisées. Elle repose également sur le fait que l'activation de ces régions entraîne une plus grande consommation d'oxygène et augmente le débit sanguin : effet BOLD. Ce sont ces informations qui sont détectées par l'IRMf en chaque point du cerveau lorsqu'il exécute une tâche particulière (motrice, sensorielle ou cognitive).

Une acquisition IRMf génère une série de 100 à 1000 images IRM. Chaque image d'acquisition est en général constituée de 32 ou 64 coupes de 64 pixels de côté. Les voxels de l'image volumique sont codés le plus souvent sur 16 bits ce qui produit des fichiers de 256 ou 512 Ko.

Les images normalisées par NMI (Normalized Mutual Information) qui permettent la comparaison entre plusieurs sujets ou l'utilisation d'atlas anatomiques sont plus volumineuses. Au final les séries IRMf complètes forment des volumes de données compris entre 25 Mo et 1 Go.
Jusqu'ici les protocoles expérimentaux (paradigmes) sont programmés de bout en bout. L'expérience est intégralement prévue au préalable. Elle passe par la conception des tâches à soumettre au cerveau ainsi que le modèle de réponse hémodynamique attendu pour chacune. Cette méthode guidée par le modèle (ang. model-driven) qui est la plus répandue à l'heure actuelle, mais elle ne peut conclure en dehors du modèle et le paradigme interdit toute intervention au cours de l'expérience.

Dans cet article, nous allons proposer une démarche pour découvrir le fonctionnement du cerveau en se basant sur un concept de fouille de données, déjà décrit dans nos publications précédentes : Korczak et al. (2005a) (2005b), ainsi que dans Goute et al. (1999) et Moller et al. (2001). Brièvement, ce concept peut se définir comme l'extraction de connaissances potentiellement exploitables à partir d'images IRMf. Il s'agit donc d'une démarche d'exploration et de découverte, radicalement différente de celle décrite préalablement. C'est une approche interactive qui intègre directement l'expert-médecin dans le processus de découverte et d'apprentissage de concepts pour mettre en évidence les zones fonctionnelles du cerveau et leur organisation.

La plateforme d'expérimentation de fouille d'images IRMf a été développée par Korczak et al. (2005) comprenant des algorithmes de classification de signaux IRMf qui permettent une fouille visuelle interactive en temps quasi réel : K-means, LBG (Lloyd généralisé), SOM de Kohonen et GNC (Gaz Neuronaux Croissants). En général ces algorithmes favorisent des groupes de forme sphérique et de tailles similaires. Ils sont très sensibles à la présence d'outliers (atypismes) dont la proximité induit l'algorithme en erreur en lui laissant laisser supposer qu'ils ont leur place au sein d'une classe.

Dans cet article, l'algorithme CURE sera décrit et comparé avec les algorithmes cités au-dessus.

CURE selon Guha et al. (1998) est un algorithme de classification, mais il est plus robuste face aux outliers et permet d'identifier des groupes non sphériques et d'une grande variance de taille. CURE réalise ceci en représentant chaque groupe par un nombre fixé de points qui sont générés en sélectionnant des points bien dispersés du groupe, et ensuite rapprochés du point moyen au centre du groupe en le multipliant par un coefficient. Le fait d'avoir plus d'un point représentatif permet à CURE de bien s'ajuster à la géométrie des clusters non sphériques et l'opération de rapprochement de ses points permet de diminuer les effets des outliers.

Pour manipuler de grands volumes de données, CURE propose une combinaison d'échantillonnage aléatoire et de partitionnement. Un échantillon tiré de l'ensemble des données est tout d'abord partitionné et chaque partition est partiellement mise en cluster. Chacun de ces groupes partiels sera à nouveau regroupé lors d'une seconde passe de l'algorithme pour extraire les clusters désirés.

L'article a été structuré en quatre sections principales. Dans la section suivante, l'algorithme CURE est décrit informellement en s'appuyant sur les extensions et les adaptations à la classification IRMf. La section 3 présente des possibilités développées dans le logiciel 3DSlicer pour la fouille d'images IRMf. La section 4 illustre les résultats de comparaison des algorithmes développés sur les données synthétiques. Elle discute aussi des avantages et des faiblesses de l'algorithme CURE.

2 Présentation de l'algorithme CURE

CURE est un algorithme de classification hiérarchique ascendante non supervisée, conçu pour traiter de grands volumes de données. Sa spécification détaillée est décrite en détail dans Guha et al. (1998). Brièvement, il débute en considérant chaque signal comme un cluster, et fusionne au fur et à mesure les deux clusters les plus proches (FIG. 1). Chaque cluster possède un ensemble de signaux représentatifs (R) qui le délimitent. Ils sont déterminés tout d'abord en choisissant les points les plus épars dans le cluster, et sont ensuite rapprochés du centre (X) par un coefficient. La distance entre deux clusters est la distance entre la paire de points représentatifs les plus proches, chacun appartenant à l'une des deux classes. Ainsi, seulement les signaux représentatifs d'un cluster sont utilisés pour calculer la distance aux autres clusters.

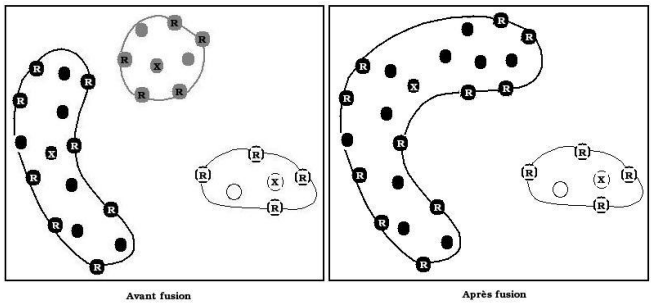

FIG. 1 - *Fusion de deux clusters*

Les signaux représentatifs tendent à s'accaparer les contours et la géométrie du cluster. En outre le rapprochement des signaux éloignés vers le centre du cluster lisse les anomalies de surface et atténue les effets des outliers qui sont généralement les signaux les plus éloignés du centre du cluster. Par conséquence, le rapprochement va obliger les outliers à se déplacer d'avantage vers le centre, pendant que les signaux représentatifs restants n'auront qu'un changement minimal à faire. Les grands mouvements des outliers devraient réduire leur capacité à indiquer les mauvais clusters à fusionner. Un traitement ultérieur pour écarter les outliers basé sur ces déplacements est envisageable. Le coefficient de rapprochement peut aussi être utilisé pour contrôler les bords du cluster, c'est à dire qu'une petite valeur rapproche peu les signaux éloignés et ainsi favorise les clusters allongés, alors qu'une grande rapproche très nettement les signaux éloignés vers le centre et les clusters tendront à être plus compacts. Comme mentionné, au départ chaque signal est considéré comme un cluster, et à chaque passe, les deux plus proches sont fusionnés en un seul. Ce processus est répété jusqu'à ce qu'il n'y ait plus que k clusters.

Au cours de la classification, chaque cluster est renseigné par son centre et l'ensemble de ses signaux représentatifs. Pour une paire de signaux p et q, $dist(p, q)$ renvoie la distance entre ces deux signaux, qui peut être une distance de Manhattan ou bien une distance euclidienne. D'autres types de distance pourraient aussi être employés. La distance entre deux clusters u et v peut être définie comme suit :

$$dist(u,v) = min\ dist(p, q) \mid p \in u\ ,\ q \in v$$

Dans notre projet, l'algorithme CURE a été adapté d'une part aux images IRMf, et d'autre part aux structures de données préexistantes dans la plateforme développée. Il se base sur deux grandes étapes que sont celle du *clustering* à proprement parler et celle de la *fusion* des dits clusters.

L'étape de *clustering* est le processus même de classification. Au départ, chaque cluster n'est constitué que d'un unique signal, qui fait donc office de centre et de signal représentatif. CURE, après avoir déterminé pour chacun quel était le cluster qui lui était le plus proche, va tous les ranger dans le tableau de clusters dans l'ordre croissant des distances à leur cluster le plus proche. Suite à cette initialisation, à chaque itération de la boucle principale, jusqu'à obtention du nombre souhaité de clusters, les deux plus proches clusters sont fusionnés par la méthode du même nom : celui qui se trouve en tête du tableau de cluster, et celui qui lui a été assigné comme plus proche. Le cluster émergeant de la fusion, se voit donc pourvu d'un jeu approprié de signaux représentatifs. L'algorithme détermine alors à nouveau pour chaque cluster, celui qui lui est le plus proche. Notamment pour ceux qui étaient en relation avec les clusters absorbés par la fusion, ceux qui ont le nouveau cluster comme plus proche voisin et enfin et surtout, pour ce nouveau cluster. Une fois cette opération terminée, les clusters, y compris nouveau, se voient rangés à leur juste place dans le tableau de clusters.

L'étape de *fusion*, consiste à regrouper les deux clusters les plus proches. Sa fonction principale est de déterminer les signaux représentatifs pour le cluster issu de la fusion. Par rapport à la version originale qui recherche les signaux représentatifs du nouveau cluster parmi tous les signaux qui le constituent, nous avons implémenté une recherche restreinte aux signaux représentatifs des clusters ayant été fusionnés.

De plus, nous déterminons de manière dynamique le nombre de signaux représentatifs pour ce nouveau cluster. Contrairement à l'algorithme original qui propose un nombre fixe de signaux représentatifs quelque soit les dimensions du cluster, nous avons opté pour un nombre variable proportionné à la dispersion de ses signaux.

Une force de CURE selon les auteurs est de pouvoir s'adapter à de grandes bases de données pour un algorithme hiérarchique. L'implémentation de la version originale a démontré certaines faiblesses de performances de la classification de signaux IRMf qui est très lourde car il s'agit de voxels auxquels s'ajoute la quatrième dimension de leur évolution dans le temps. Pour réduire le temps de classification, nous avons appliqué quelques améliorations.

– *Tirage aléatoire.* Un tirage aléatoire des données est utilisé ayant pour vertu d'améliorer la qualité de la classification. En effet, les signaux sont enregistrés selon l'ordre dans lequel l'IRM les balayent, ce qui fait que deux signaux qui sont issus de zones voisines dans le cerveau, peuvent être séparés lors de l'enregistrement car une couche est balayée dans un sens avant de passer à la couche inférieure.

– *Echantillonnage.* Cela permet de déterminer les classes, avec moins de signaux. La taille de l'échantillon est paramétrable et pourrait donc être modifiable par le médecin. Ce cas est extrêment important car CURE fonctionnant de manière hiérarchique, plus le nombre de signaux est important, plus il génère de classes au départ et plus les calculs entre toutes les classes prennent du temps et des ressources.
– *Partitionnement.* Sur cette même constation, un système de rechargement en signaux a été réalisé. Ici encore la taille du partitionnement pourrait être laissé à la discrétion du médecin. CURE classant les clusters par ordre croissant de leur distance au cluster qui leur est le plus proche, impose donc un calcul de distance entre chaque paire de clusters, et pour chaque paire, leur distance est la distance minimale entre tous les couples de signaux représentatifs des deux classes. Nous avons déterminé expérimentalement un nombre de clusters seuil au delà duquel l'algorithme est trop ralenti. Pas à pas l'algorithme fusionne deux à deux les clusters jusqu'à atteindre une valeur plancher à partir de laquelle nous effectuons un rechargement en nouveaux clusters pour réatteindre le nombre maximal fixé. Ce procédé est répété jusqu'à épuisement du nombre de signaux.

Les outliers sont des sources de perturbation importante des algorithmes de classification. Dans notre cas, les clusters de petite cardinalité peuvent se trouver être des clusters d'un intérêt important, alors que d'autres qui sont très nombreux comme ceux comportant des signaux émis par la matière blanche ou le liquide céphalo-rachidien sont d'une inutilité flagrante. La présence du médecin expert et sa faculté à pouvoir intervenir lors du processus, permet de s'abstenir pour l'instant de ce genre de traitement de suppression de ces signaux intrus, puisqu'il peut le faire lui-même en pleine connaissance de cause.

L'implémentation dans la plateforme de telles modifications rendent CURE plus efficace et robuste à la classification des données qui sont lourdes, et nombreuses. Nous avons pour conséquence, des temps de calcul amoindris et grâce au tirage aléatoire une meilleure classification.

3 Fouille visuelle d'images IRMf

L'algorithme CURE a été inclus dans un outil de visualisation 3DSlicer (www.slicer.org), développé par l'école de médecine d'Havard en collaboration avec le MIT. 3DSlicer a été conçu principalement pour la visualisation temps réel au cours d'une intervention chirurgicale sur le cerveau. Il peut présenter cinq vues différentes du cerveau, trois coupes orthogonales, une vue 3D d'ensemble et une vue 3D dite endoscopique pour naviguer à l'intérieur du volume (cette dernière n'apparaît pas sur l'exemple et n'a de toute façon pas d'intérêt pour l'IRMf).

Elles peuvent combiner et afficher simultanément 3 volumes qui n'ont pas nécessairement la même résolution. La vue 2D affiche les trois coupes et les labels modélisés sous forme de surface. En dehors de l'algorithme CURE, nous avons implémenté les fonctions de 3DSlicer permettant

l'affichage des résultats dont notamment la partie gauche qui permet de renseigner les paramètres de calcul et d'afficher des résultats tels que la forme du signal du cluster en cours. Sur la figure 2 les clusters apparaissent de couleurs différentes et cohérentes avec la liste déroulante de la fenêtre de gauche, ainsi que les statistiques affichées en temps réel durant le processus de classification.

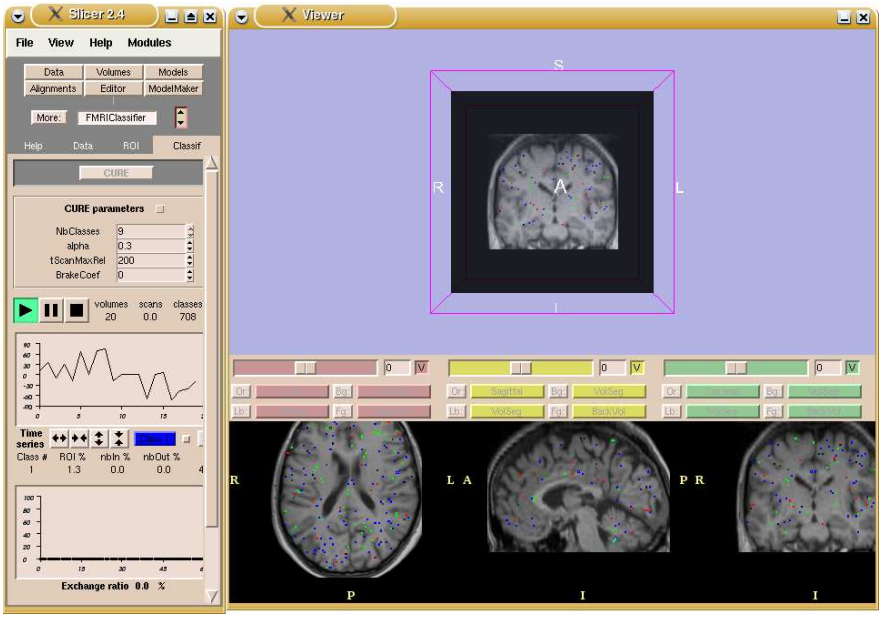

FIG. 2 - *Interface de contrôle FMRIClassifier dans 3DSlicer*

Durant la classification, le médecin peut changer facilement de point de vue avec la souris et modifier le contenu de l'affichage. Pour l'aspect interactif, il est pourvu de deux points forts. Le premier est qu'il est utilisé pour visualiser par IRM en temps réel, ce qui garantit une très bonne réactivité. Deuxièmement, 3DSlicer affiche les résultats des algorithmes à mesure de leur découverte. C'est par cette interface permettant la visualisation 3D que l'expert médecin peut également intervenir. Il peut mettre la procédure en pause pour modifier certains paramètres, focaliser sur son centre d'intérêt, éliminer les classes indésirables comme les yeux par exemple. L'interface du module de classification inclus dans 3DSlicer a été détaillée par Korczak (2005b).

4 Résultats d'expérimentation

Dans cette étude nous avons voulu comparer CURE aux autres algorithmes déjà implémentés. Ceux-ci avaient déjà subit des protocoles comparatifs par Hommet (2005). Nous avons donc repris

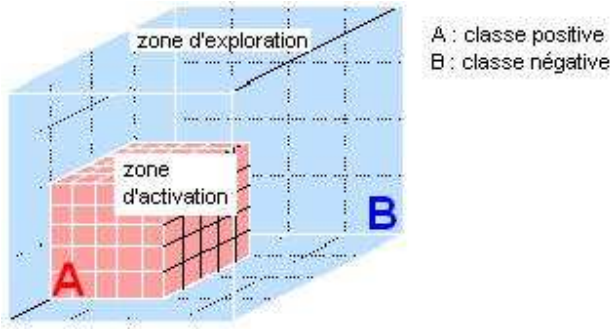

FIG. 3 - *Schéma simplifié des classes d'activation positives et négatives*

les mêmes démarches permettant de comparer leurs performances de détection des zones d'activation. Travailler sur des données réelles impliquerait d'avoir un outil permettant de nous indiquer que l'algorithme a tort ou raison. La station d'acquistion et de fouille d'images IRMf en temps reel n'existe pas, donc dans nos évaluations des algorithmes nous avons reformulé les expérimentations effectuées sur les benchmarks de SPM (Statistical Parametric Mapping, http ://www.fil.ion.ucl.ac.uk /spm), qui sont connus comme la référence dans le domaine. Il s'agit d'un assemblage de deux parties, les activations simulées purement artificielles et le fond constitué de données réelles. La série d'IRMf du cerveau utilisée comme fond a été composée à partir d'images issues du test auditif SPM. Le test auditif est un enchaînement de deux conditions : "silence" et "parole". Toutes les images de la condition "silence" ont été rassemblées et enchaînées pour former le fond de notre jeu de test qui se rapproche ainsi au mieux de données réelles.

Sur cette série de 40 images, viennent s'ajouter les activations synthétiques formées par des séries temporelles en créneau (signal carré) simulant un paradigme de type bloc. La zone d'activation est un volume cubique de 5 voxels de côté. Le niveau de bruit moyen de ces 125 voxels du fond a été mesuré en prenant le double de la variance des intensités de ces voxels dans le temps. A partir de cette mesure, il est ensuite possible de contrôler le rapport signal sur bruit de nos activations en ajoutant des créneaux d'intensités voulues aux voxels considérés. La figure 3 illustre de manière simplifiée comment sont déclarées les classes positives en fonction de leur recouvrement d'une zone d'activation.

En utilisant des données synthétiques, les algorithmes ont été comparés en variant différents critères qui avaient été pris en considération sur les précédents tests, à savoir : le nombre de classes, le rapport signal sur bruit et le rapport de dillution de la zone d'activation dans la zone d'exploration.

La qualité de la classification est proportionnelle à la faculté de l'algorithme à retrouver les bonnes classes.

Ainsi pour être déclarée positive, une classe doit avoir tous ses signaux qui appartiennent à la zone activée. Une classe négative ne possède pas de signaux dans cette zone. Le point critique intervient lorsqu'une classe se trouve à cheval sur ces deux définitions.

Un critère nous permet de déterminer dans quelle catégorie ranger la classe :

$$C_a \text{ positive} \equiv (inC_a / outC_a) > (2nZA / ((nZE - nZA) / (nbClasses - 1)))$$

où *nZE* est le nombre de voxels dans la zone explorée ; *nZA* est le nombre de voxels dans la zone activée ; *nbClasses*, le nombre total de classes ; C_a, une classe ; *et* inC_a, $outC_a$, le nombre de voxels de C_a à l'intérieur et à l'extérieur de la zone activée. Ce critère a été repris des tests effectués sur les algorithmes préimplémentés sur la plateforme afin de pouvoir les comparer sur les mêmes bases. Les tests consistent à soumettre aux algorithmes les séries constituées et à comparer les résultats de classification avec les résultats attendus. Ils ont été effectués de manière automatique et répétés 25 fois pour estimer d'une part l'erreur réelle et d'autre part pour voir la convergeance des algorithmes. Les autres valeurs ont été choisies dans les limites (intervalles) définies par des experts médecins. La moyenne des essais constitue alors le résultat définitif d'un test et renseigne sur la fréquence de détection de la zone activée.

Les tableaux 1, 2 et 3 présentent les résultats obtenus aux côtés des évaluations précédentes des autres algorithmes. Les classifications ont été réalisées par variation respective des paramètres que sont le nombre de classes (*nC*), le rapport de dilution des voxels activés (*nZA/nZE*) et le rapport signal sur bruit (*S/B*). Les résultats sont exprimés en pourcentage de détection de la zone d'activation.

Un algorithme est d'autant meilleur qu'il permet de retrouver des zones actives même si elles sont peu nombreuses en rapport du nombre total de voxels. Et aussi qu'il ne se laisse pas trop influencer par les bruits parasitants les signaux.

nC	4	6	8	9	10	12	14	15	16	18	20	21	24
GNC	10	0	95	100	100	100	100	100	100	100	100	100	100
SOM	4	24	56	40	52	70	100	98	100	100	100	100	100
LBG	2	26	36	30	36	54	64	72	88	96	100	96	100
K-Means	9	16	36	40	44	40	60	58	64	84	68	70	84
CURE	56	60	64	68	76	85	80	80	68	72	76	80	84

TAB. 1 - *Fréquence de détection (%) en fonction du nombre de classes imposé nC*

nZA/nZE	1/343	1/216	1/125	1/64	1/27	1/8
GNC	0	85	100	100	100	100
SOM	0	15	40	100	100	100
LBG	10	34	30	95	100	100
K-Means	10	35	40	45	95	100
CURE	52	72	68	68	72	72

TAB. 2 - *Fréquence de détection (%) en fonction du rapport nZA/nZE*

S/B	1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	2,0	3,0	4,0
GNC	0	0	55	95	100	100	100	100	100	100	100	100	100
SOM	0	0	0	0	0	40	70	85	100	100	100	100	100
LBG	0	5	0	15	15	30	55	40	65	75	69	100	100
K-Means	6	1	15	35	15	40	44	70	50	55	70	82	90
CURE	52	64	52	68	76	68	72	80	72	68	72	88	84

TAB. 3 - *Fréquence de détection (%) en fonction du rapport signal sur bruit S/B*

Le choix de ces paramètres a été discuté avec les experts médecins. Le nombre de classes qui est généralement la condition d'arrêt des algorithmes permet de juger d'une bonne valeur par défaut à fournir au médecin, et par la suite savoir dans quelle fourchette de valeur, l'algorithme s'égare. La dillution des voxels activés dans les voxels explorés permet de savoir si l'algorithme peut retrouver des classes de faible cardinalité. En effet, notre cerveau n'utilise qu'une petite partie de ses neuronnes pour certaines tâches. Si le médecin désire trouver ces petites classes il est bon de pouvoir lui recommander un algorithme plutôt qu'un autre. Enfin le rapport signal sur bruit est capital à cause de la présence de signaux parasites comme le mouvement des yeux ou les bruits émis par l'IRM.

Il apparaît que CURE présente des résultats meilleurs que les autres algorithmes pour les tests avec un nombre de classes inférieur à 8. Ces performances augmentent avec le nombre de classes mais de manière plus faible que les autres algorithmes. CURE présente une alternative intéressante dans les cas de forte dillution (faible pourcentage de zones actives *nZA/nZE*) et de signal sur bruit très faible. En comparaison avec le meilleur algorithme testé, CURE offre des performances médiocres. Il est à noter cependant que ces données synthétiques n'exploitent pas les qualités d'adaptation de CURE aux classes non sphériques et de forte variance.

Notons que ces résultats sont issus des algorithmes tels qu'ils sont implémentés à l'heure actuelle, autrement dit en mode batch. Le déroulement de l'algorithme est pausé régulièrement pour lui permettre une communication avec l'interface visuelle 3DSlicer, mais cela ne modifie pas la classification. Une prochaine étape qui s'inscrit dans le cadre de l'interactivité avec l'expert médecin serait de remodeler les algorithmes pour leur permettre un déroulement autre que batch. Le concept serait de pouvoir réserver une classe qui nous semble pertinente à un moment donné de la classification. En la figeant, l'algorithme ne pourrait plus interagir avec elle pour lui ajouter des signaux ou la fusionner à d'autres. Cela améliorerait grandement les résultats de classification, car le médecin sachant identifier la sémantique des classes, pourra empécher qu'une classe intéressante soit altérée d'une part, mais également que d'autres classes puissent être perturbées par une fusion malencontreuse avec celle-ci. De plus, des règles de classification pourront ainsi progressivement se dégager pour faire tendre la classification elle-même à la détection de ces classes.

5 Conclusion

Dans cet article, un algorithme hiérarchique de classification non supervisée a été présenté et validé sur des images IRMf. L'algorithme développé est une extension d'algorithme CURE proposé par Guha et al. (1998) pour traiter des volumes de grande taille en temps quasi réel.

Les extensions effectuées se portent sur les domaines suivants que sont le tirage aléatoire des signaux, leur partitionnement, leur échantillonnage ainsi que la représentativité des clusters. En tant qu'algorithme hiérarchique, CURE est extrêment gourmand en ressources. Nos améliorations ont réduit la complexité de l'algorithme notamment en limitant le nombre de calculs de distance et en conséquence ont réduit les temps de calculs. Selon la simulation on peut envisager une utilisation d'algorithme CURE étendue avec des contraintes de temps réel. L'intégration de cet algorithme dans la plateforme 3DSlicer facilite l'interaction d'un expert médecin dans le processus d'analyse d'images IRMf. L'algorithme CURE a été testé sur des données simples bi-dimmensionnelles et sur des données synthétiques. Si sur les premières, CURE obtient une très bonne performance, cependant il s'avère que sur les secondes, il présente des performances moyennes, mais reste de bonne robustesse. Cette constatation ne concerne que des données synthétiques ne lui permettant pas de mettre en avant ses qualités d'adaptation à des clusters d'une morphologie non sphérique.

La plateforme de fouille d'images IRMf et l'algorithme CURE dans la version actuelle nécessitent des validations de grande envergure par des médecins spécialistes. Pour l'instant les algorithmes classent les signaux, mais n'ont aucune mémoire de leurs précédents classements, ils ne possèdent donc aucune expérience. Ce travail est d'une très grande ampleur et nécessitera de longs mois ou années de travail pour aboutir. En effet, l'objectif futur sera l'intégration de connaissances médicales préalables dans l'algorithme, ainsi que l'acquisition des connaissances issues de la classification, dans une base de connaissances au niveau du patient pour suivre sa propre évolution, mais aussi au niveau de tous les sujets traités pour les comparer et tirer des enseignements médicaux, et des conseils lors des prochaines fouilles.

Remerciements

Les auteurs remercient Christian Scheiber (IHC, Lyon) pour les données expérimentales et Jean Hommet, Nicolas Lachiche, LSIIT, Illkirch, pour les conseils et aides dans la réalisation de ce projet.

Références

C. Goute, P. Toft, E. Rostrup, F.A. Nielsen, et A.K. Hansen (1999). *On clustering FRMI time series.* NeuroImage, 9, pages 2398-3100.

S. Guha, R. Rastogi, K. Shim (1998). *CURE : An Efficient Clustering Algorithm for Large Databases*. SIGMOD 1998, pages 73-84.

J. Hommet (2005). *Fouille interactive de séquences d'images 3D d'IRMf*. Rapport de LSIIT, CNRS, Illkirch.

J. Korczak, C. Scheiber, J. Hommet, N. Lachiche (2005a). *Fouille interactive en temps réel de séquences d'images IRMf.* Numéro Spécial RNTI.

J. Korczak, C. Scheiber, J. Hommet, N. Lachiche (2005b). *Exploration visuelle d'images IRMf basée sur des Gaz Neuronaux Croissants*. Atelier sur la fouille de données complexes, EGC 2005, Paris.
U. Moller, M. Ligges, C. Grunling, P. Georgiewa, W.A. Kaiser, H. Witte et B. Blanz (2001). *Pitfalls in clustering of neuroimage data and improvements by global optimization strategies*. NeuroImage, 14, pages 206-218.
SPM. Statistical Parametric Mapping. *Welcome Department of Imaging Neuroscience*. http ://www.fil.ion.ucl.ac.uk/spm.

Summary

The human brain provides typical complex data for data mining. Extracting active voxels from brain images is often very difficult due to a very high level of various noises. First experiments of current data mining algorithms in this domain showed their low performances and recognition abilities. In this article, an extended unsupervised data mining algorithm CURE is described and evaluated. CURE is compared with several unsupervised algorithms on fMRI images, reporting results with respect to the number of classes, the noise level, and the ratio of the activated/observed areas.