MVM, un Modèle de Vision Multiéchelle

(version 1.0a)

Documentation utilisateur

DRAFT NON FINALISÉ

Juillet 2003

Myriam Traina
Eric Slezak
Albert Bijaoui

Observatoire de la Côte d'Azur

Table des matières

1 Introduction **1**
1.1 Le concept d'une vision multiéchelle 1
1.2 Les différents programmes du logiciel MVM 4
1.3 Historique du logiciel MVM 8

2 Calcul de la transformée et sélection des coefficients significatifs **9**
2.1 Les étapes du programme Thresh–wave 9
2.1.1 Transformation en ondelettes de l'image de départ . . . 9
2.1.2 Identification des coefficients significatifs 10
2.2 **Syntaxe du programme thresh–wave** 13
2.3 Exemple d'application 14

3 Segmentation des régions significatives et graphes interéchelles 19
3.1 Segmentation et sous-segmentation des plans en ondelettes seuillés 19
3.2 Etablissement des connections interéchelles 20
3.3 **Syntaxe de la procédure connect–wave** 21
3.4 Exemple d'applications 22

4 Identification de structures **25**
4.1 Etapes du programme **obj–wave** 25
4.2 **Syntaxe de la procédure** *obj–wave* 26
4.3 Exemple 27

5 Reconstructions **29**
5.1 La méthode du gradient conjugué 29
5.2 Reconstruction à partir des plans seuillés 30
5.2.1 Le programme *rec–image* 30
5.2.2 **Syntaxe de la procédure rec–image** 30
5.2.3 Exemple 32

5.3 Reconstructions à partir d'arbres 32
5.3.1 **Syntaxe de la procédure rec–obj** 32
5.3.2 Exemples 34

6 Le programme *mvm–rec* **37**
6.1 Fichier de configuration 37
6.2 Paramétrisation des objets 41
6.3 Limitations du logiciel MVM 1.0a 43

7 Applications 2D **45**
7.1 Reconstructions de profils photométriques typiques 45
7.1.1 Stabilité de la reconstruction selon la procédure utilisée 45
7.1.2 Qualité des reconstructions : Cas de la loi de Gauss . . 49
7.1.3 Qualité des reconstructions : Cas des autres lois 59
7.1.4 Reconstruction de plusieurs profils 69

Conclusions et Perspectives **i**

Annexes **i**
7.2 Algorithme à trous i
7.3 Outils de visualisations du graphe des structures identifiées . . i
7.3.1 ***voir–foret*** i
7.3.2 complexe ii
7.3.3 ***carte–obj*** ii
7.3.4 ***dessine–arbre*** ii
7.3.5 ***make–liste*** iv
7.4 Evaluation morphologique et photométrique : Catalogue d'objets v

Chapitre 1

Introduction

Si l'on exclue le cas d'une collection d'objets identiquement égaux à la réponse impulsionnelle du détecteur (ex. : l'image d'un amas globulaire), les objets et images astronomiques profondes se caractérisent par des formes floues d'aspect et de taille très variables ainsi qu'un encombrement grandissant qui voit la superposition d'objets astrophysiques différents ne plus être une exception. De là des difficultés grandissantes pour produire un catalogue complet et fiable de tous les objets présents dans de telles images. En effet, les différents logiciels d'analyse mis en place au cours des années quatre-vingt ont tous été développés avec comme point de départ l'extraction et la mesure d'objets bien isolés se détachant sur un fond clairement identifiable. Les limitations inhérentes à l'approche sont connues : introduction d'une échelle privilégiée pour calculer la carte des variations basse-fréquence dans l'image, autrement dit le fond, définition univoque des objets par une bijection rigoureuse avec les domaines connexes issus d'une segmentation d'image. Les améliorations qui peuvent être apportées à ce modèle (correction de la carte, série de seuils) ne résolvent les difficultés citées que partiellement. Devant cet état de fait, les bases d'un nouveau modèle de vision ont été posées il y a maintenant près de dix ans (Bijaoui & Rué 1995). L'objet de ce document est d'en rappeler le concept clef et de présenter les différents programmes d'analyse construits autour depuis.

1.1 Le concept d'une vision multiéchelle

Introduite à la fin des années quatre-vingt (cf. Meyer 1992), la transformation en ondelettes permet de mesurer l'importance *locale* des différentes fréquences dans un signal quelconque par le biais de ses convolutions avec une série de filtres se déduisant les uns des autres par dilatation. Les pro-



FIG. 1.1 – Image de la galaxie M83

priétés de ces filtres sont en effet telles que l'opération permet de percevoir d'un côté ce signal à différentes résolutions (filtres passe-bas) et de l'autre les détails (filtres passe-bande) qui apparaissent ou disparaissent lors du passage d'une approximation du signal à une autre (Mallat 1989) ; le poids à chaque échelle de ces détails dans la constitution du signal est codée par l'amplitude des coefficients en ondelettes de la transformée. La figure 1.2 visualise la transformée de la galaxie M 83 sur les huit premières échelles d'une gamme logarithmique. On remarque aisément que les éléments formant cet objet structuré y sont perçus chacun à l'échelle la mieux adaptée à leur taille typique. Cette capacité des ondelettes à déployer le contenu fréquentiel de chaque pixel est au cœur du modèle dont il est question ici.

Ce déploiement de l'information sur l'axe des échelles permet en effet de décrire naturellement toute organisation hiérarchique, qu'elle soit le fait d'un objet intrinsèquement structuré ou qu'elle provienne de la juxtaposition fortuite en un ensemble composite de plusieurs entités relativement simples par elles-mêmes. La figure 1.3 en offre une illustration frappante. Le contour associé à grande échelle à toute la région de la la nébuleuse NGC 40 s'affine en se ramifiant à mesure que l'augmentation de la résolution fait accéder aux détails plus fins dans l'image, qu'il s'agisse de structures de plus petites tailles que l'enveloppe globale, d'une étoile accolée à la nébulosité ou du bruit. On peut dès lors penser à juste titre qu'il est beaucoup plus facile de séparer dans ce nouvel espace le signal du bruit, de distinguer ses différentes composantes structurales, voire de les associer ou dissocier selon les besoins ou la nécessité, toutes tâches habituellement délicates ; il suffit de réussir à isoler les régions intéressantes avant de repasser dans l'espace direct par trans-

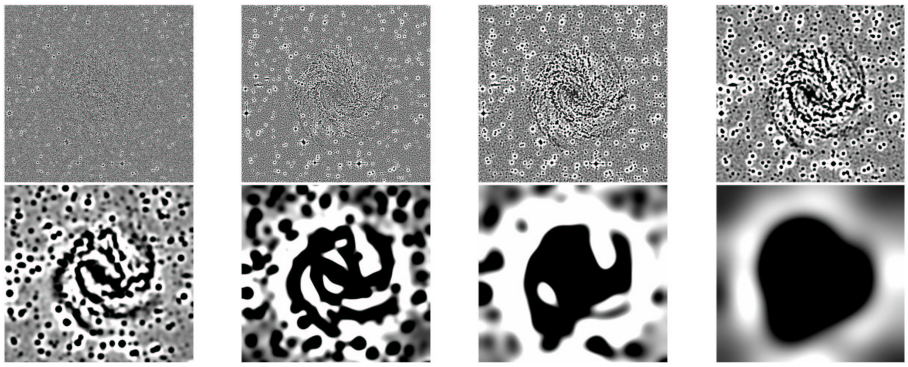


FIG. 1.2 – Transformée en ondelettes de l'image de la galaxie M 83 par un algorithme "à trous" avec comme fonction d'échelle une fonction B-spline cubique. Seules sont visualisées les huit premières échelles allant d'un facteur de dilatation $a = 2^0$ à $a = 2^7$. On constate que les détails formant l'objet sont préférentiellement mis en lumière à l'échelle qui correspond le mieux à leur taille caractéristique.

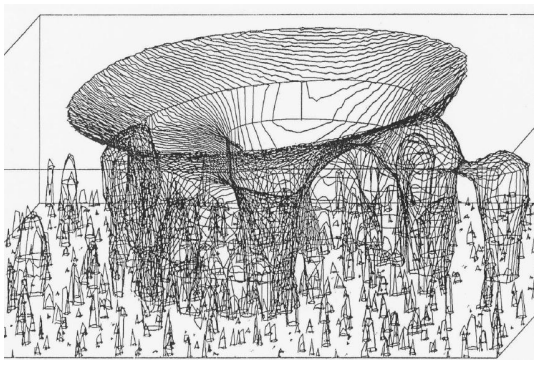


FIG. 1.3 – Visualisation d'une surface d'égale valeur pour les coefficients en ondelettes dans le volume de la transformée de la nébuleuse NGC 40. L'organisation hiérarchique de l'information est clairement visible depuis une vision globale de l'objet jusqu'aux détails les plus fins constituant le signal.

formation inverse. Aussi le nouveau modèle de vision multiéchelle (MVM) proposé repose-t-il sur ce concept d'une vision arborescente des éléments définissant l'image astronomique dans l'espace de sa transformée en ondelettes. Le bruit et le fond n'y sont que deux composantes parmi d'autres à détecter aux échelles qui leur correspondent tandis qu'à toute stucture est associée un certain volume de coefficients en ondelettes, cette fraction de la transformée pouvant n'être elle-même qu'un embranchement du sous-ensemble plus vaste lié à la contrepartie d'une éventuelle structure de plus grande taille contenant la première et ainsi de suite.

1.2 Les différents programmes du logiciel MVM

Le propos du logiciel MVM est de fournir à terme tous les moyens pour analyser correctement une image astronomique avec des structures emboîtées. L'exemple le plus simple est celui de la superposition d'un bruit blanc à une structure quelconque avec comme difficulté de réussir à préserver les détails les plus fins du signal lors de l'élimination du dit bruit. Beaucoup plus complexe est la séparation optimale de formes floues enchevétrées avec des brillances très différentes. Dans les deux cas, le but commun recherché est une image correcte de chacune des structures jugées intéressantes afin que les évaluations qui suivent soient les plus justes possibles. Pour atteindre cet objectif dans le cadre du modèle de vision multiéchelle des objets qui sous-tend le logiciel MVM, la méthodologie adoptée fait intervenir les cinq étapes qui suivent :

a) un déploiement sur la gamme d'échelles définie par une transformation en ondelettes de l'information contenue dans l'image à différentes résolutions spatiales ;

b) une sélection des coefficients en ondelettes liés au signal en rejetant à chaque échelle ceux issus des fluctuations du bruit présent dans l'image ;

c) la mémorisation de l'emboîtement local des régions associées au signal aux différentes échelles ;

d) la recherche de la contrepartie des structures de l'image dans l'espace de la transformée avec détermination de leur échelle caractéristique et de l'arborescence de coefficients qui leur est associée ;

e) un retour dans l'espace direct par l'élaboration d'une image compatible avec la hiérarchie de coefficients en ondelettes qu'il a été décidé d'isoler.

Ces étapes méthodologiques sont codées dans un certain nombre de programmes présentés ci-après et détaillés dans les chapitres qui suivent.

i) *thresh–wave :* Ce programme démarre le traitement de l'image à analyser en réalisant deux opérations. La première est la transformation en ondelettes de l'image sur un ensemble d'échelles dyadiques. Cela est le fait d'un algorithme sans décimation dit *à trous* avec pour fonction d'échelle une fonction B-spline d'ordre trois. La seconde est la sélection des coefficients en ondelettes positifs dont la valeur correspond à une probabilité suffisamment faible d'être la conséquence du bruit présent dans l'image (processus de Gauss, de Poisson, composite). Cette sélection repose sur un test binaire (*hard thresholding*) vis-à-vis d'une valeur critique qui dépend de l'échelle considérée et du risque de fausse alarme toléré.

ii) *thresh–abs–wave :* Il s'agit d'une variante du programme précédent où le test statistique porte sur la valeur absolue des coefficients en ondelettes. La transformée seuillée se présente donc sous la forme d'une série de plans en ondelettes, un par échelle explorée, contenant tous les coefficients jugés significatifs qu'ils soient positifs ou négatifs. La valeur nulle est affectée aux autres pixels.

iii) *connect–wave :* L'exécution de ce code permet de regrouper localement les coefficients positifs significatifs le long de l'axe des échelles spatiales. L'objectif est d'isoler les coefficients issus d'une même région de l'image de départ au sein d'une entité individualisée, cela afin de pouvoir traiter chacune de ces entités selon le mode souhaité. Les pixels non nuls des plans en ondelettes seuillés sont donc tout d'abord rassemblés en domaines connexes au moyen d'une segmentation d'image ; il est possible de la faire suivre d'une sous-segmentation pour être certain de n'avoir qu'un seul maximum local par sous-domaine. Les connections inter-échelles entre les domaines (ou sous-domaines) des différents plans sont ensuite établies en examinant de façon récursive si la position de leur maximum local se projette ou non à l'intérieur d'un domaine de l'échelle immédiatement supérieure. Chaque série de liens est mémorisée avec les pixels des domaines mis en relation dans une entité appelée *arbre* au sein duquel les domaines ou sous-domaines sont dénotés *éléments*.

iv) *obj–wave :* Ce programme identifie les éléments dont le coefficient en ondelettes maximum est supérieur à ceux des autres éléments qui lui sont reliés dans l'arbre dont il fait partie. Cette réponse maximale le long de l'axe des échelles est évaluée au choix dans $L1$ (amplitude du coefficient) ou dans $L2$ (son énergie). Elle signe la présence à cette échelle et à cet endroit d'une structure statistiquement pertinente dans l'image

de départ. À chacun de ces éléments particuliers dénommé *structure* est affecté un sous-ensemble plus ou moins important de la hiérarchie des éléments de l'arbre. Ce sous-ensemble définit ce qui est appelé un *graphe des structures* ou par abus de language un *graphe d'objet*.

v) *rec–image* : Ce programme permet de reconstruire une image dans l'espace direct à partir des coefficients des seuls plans en ondelettes seuillés, sans donc que les notions d'arbre ou de structure pertinente aient à intervenir. La différence entre les coefficients en ondelettes des régions significatives et ceux des mêmes régions dans l'image reconstuite est minimisée à l'aide d'un algorithme itératif de gradient conjugué.

vi) *rec–image–residu* : Programme identique à *rec–image* à la nuance qu'il prend également en compte les coefficients négatifs significatifs dans la minimisation de l'écart entre le jeu des coefficients significatifs de l'image traitée et celui de la solution élaborée.

vii) *rec–obj -A* : L'emploi de ce programme permet d'obtenir l'image reconstruite comme la juxtaposition des imagettes obtenues pour chaque arbre à partir de l'ensemble des coefficients mémorisés sous forme d'éléments.

viii) *rec–obj* : En l'absence de l'option *-A*, le programme *rec–obj* fournit la juxtaposition des imagettes obtenues cette fois ci pour chaque structure à partir de la hiérarchie d'éléments qui lui est associée au sein de son arbre.

Ces différents programmes peuvent être enchaînés selon trois procédures représentées figure 1.4. Elles correspondent à des traitements de complexité croissante dont l'intérêt est explicité dans le chapitre intitulé "Applications". La première de ces procédures se limite à une transformation en ondelettes suivie de la sélection des coefficients significatifs et d'une reconstruction à partir des plans seuillés. Elle permet de débruiter à moindre coût une image bidimensionnelle au format FITS. S'appuyant sur la création des arbres, la deuxième permet d'avoir une qualité de reconstruction optimale pour chaque imagette issue de ces arbres et de décider ou non de les intégrer dans l'image finale. La troisième pousse l'analyse de la transformée jusqu'à l'identification des structures pertinentes dans l'image. Elle offre ainsi la possibilité d'obtenir une image finale avec uniquement certaines d'entre elles par le biais d'une reconstruction ciblée qui peut être accompagnée d'un catalogue de mesures (cf. annexes). Accompagnant cette stratégie, un certain nombre d'outils de visualisations sont disponibles dans leur version de développement (cf. annexes).

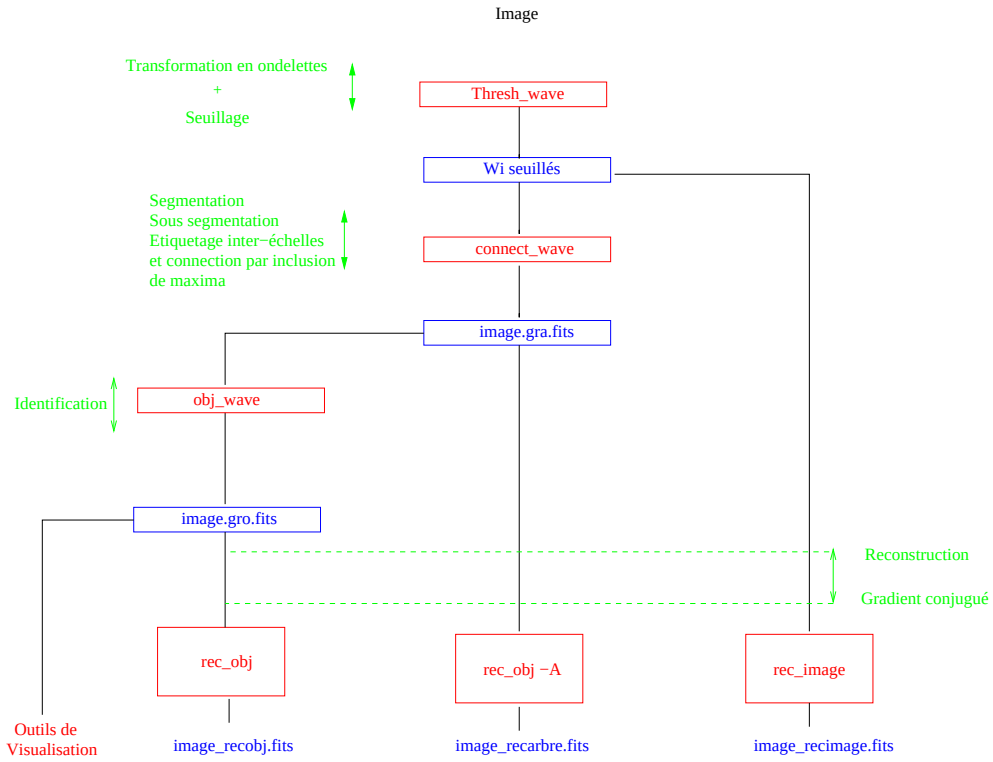


FIG. 1.4 – Les principales procédures d'analyse

- en bleu : les entrées/sorties
- en rouge : les procédures
- en vert : les descriptions

1.3 Historique du logiciel MVM

Chapitre 2

Calcul de la transformée et sélection des coefficients significatifs

2.1 Les étapes du programme Thresh–wave

2.1.1 Transformation en ondelettes de l'image de départ

La transformation en ondelettes permet de déployer l'information d'un signal sur différentes échelles : les petits détails apparaissent préférentiellement aux petites échelles de la transformée tandis que les structures étendues seront plutôt localisées aux grandes échelles. La fonction d'échelle employée est une fonction B-spline cubique qui conduit à une analyse quasi isotrope de l'image similaire à celle qu'offrirait les différences entre des lissages gaussiens de plus en plus forts. Cette solution qui ne privilégie aucune direction particulière est plus que satisfaisante pour une première exploration du signal si celui-ci ne présente pas de structures excessivement alongées.

La transformée sur une gamme d'échelle dyadique $a = 2^{i-1}$ est obtenue au moyen de l'algorithme à trous décrit par exemple dans Bijaoui *et al.* (1994). Le nombre d'opérations intervenant dans les convolutions successives de ce schéma récurrent reste constant quelle que soit l'échelle a grâce aux "sauts" introduits dans le calcul. Il en découle une grande rapidité d'exécution. Aucune décimation n'est appliquée, d'où un nombre de pixels identique sur chacun des plans de la transformée d'une image bidimensionnelle. Cette redondance assure l'invariance par translation. Elle permet par ailleurs de comparer aisément les détections d'une échelle à l'autre et facilite donc le passage entre les représentations d'une structure dans l'espace direct et dans l'espace de la transfomée.

Quatre solutions sont proposées pour gérer les effets de bords qu'induisent inévitablement des convolutions par des filtres symétriques. Les options offertes sont de les minimiser par :
– continuité ; dans ce cas l'image est étendue vers l'extérieur en reproduisant la valeur du pixel de bord. Les pixels d'indice $(-3; -2; -1)$ sont ainsi pris égaux au premier pixel de l'image d'indice 0.
– miroir, auquel cas les extensions de l'image proviennent de la réflection sur ses bords de la valeur des pixels. À titre d'exemple, un des bords étant donné par l'indice 0, les pixels d'indice $(-3; -2; -1)$ ont par cette entremise les mêmes valeurs que les pixels d'indice (3, 2, 1).
– repliement, où la topologie de l'image est prise comme étant celle d'un tore par identification des bords opposés deux à deux. La suite d'indices $(-3; -2; -1; 0; 1; 2; ...; N-2; N-1; N)$ s'identifie dans ce contexte à la suite $(N-2; N-1; N; 0; 1; 2; ...; N-2; ...)$.
– plongement, où la valeur des pixels de l'image au delà des bords est obtenue à partir d'une loi de distribution gaussienne de moyenne et d'écart-type fixés par l'utilisateur.

Lors de l'exécution du programme *thresh–wave*, seuls les plans en ondelettes seuillés (voir ci-après) sont sauvegardés sur disque. Si l'on souhaite disposer des plans dans leur intégralité, il faut exécuter le programme *trans–wave* qui ne fait que calculer la transformée d'une image (plusieurs algorithmes sont possibles : "à trous", pyramidal, multirésolution).

2.1.2 Identification des coefficients significatifs

L'analyse de l'image suppose d'être capable d'isoler le signal pertinent du bruit qui tend à le brouiller. Dans l'espace de la transformée, cela suppose d'être capable de savoir si la valeur d'un coefficient en ondelettes est suffisante pour rejeter l'hypothèse qu'il provient du bruit. Dans l'affirmative ce coefficient sera considéré comme statistiquement significatif et conservé. Dans la négative, il sera écarté.

Cette sélection des coefficients selon leur probabilité d'être associé au bruit présent dans l'image de départ s'effectue par l'intermédiaire d'un test vis-à-vis d'une valeur critique. L'étude des caractéristiques du bruit permet de connaître les valeurs de ce seuil selon l'échelle considérée. La procédure mise en place pour l'instant permet de gérer un bruit additif non corrélé spatialement de nature soit gaussien soit poissonien, soit encore une combinaison des deux (ex. : caméra CCD). Dans ces deux derniers cas, une transformation d'Anscombe (Anscombe 1948) préalable permet de stabiliser la variance sur

l'image et de se ramener à la stratégie suivie pour le cas d'une loi de Gauss stationnaire. Quelle est-elle ?

Seules des opérations linéaires étant réalisées lors de la transformation en ondelettes, cette stratégie s'appuie sur le fait que la distribution des coefficients d'un bruit blanc gaussien stationnaire est également une fonction gaussienne. L'écart-type σ de cette gaussienne diminue avec l'échelle puisque les variations d'intensité sont indépendantes d'un pixel à l'autre ; le facteur de décroissance entre deux échelles successives de la gamme dyadique vaut théoriquement 0,5 pour un signal bidimensionnel et une base orthogonale d'ondelettes. Il suffit donc de pouvoir accéder à la valeur de la variance à l'échelle examinée pour connaître la probabilité qu'a toute valeur d'apparaître du simple fait des fluctuations d'intensité inhérentes au bruit. La pertinence d'un coefficient issu d'un signal bruité est alors donnée par la simple comparaison de sa valeur à celle correspondant au risque de fausse détection que l'on s'autorise. Rappelons que le choix d'un test à quatre ou trois σ implique un risque de 0,03 et de 1,35 pour mille, soit respectivement plus de 30 et 1400 fausses alarmes à l'échelle 1 pour une image de taille 1024×1024.

L'estimation de la variance induite par le bruit dans l'image traitée sur les coefficients de l'échelle $a = 1$ est effectuée par le biais d'une statistique sur les coefficients de cette image à cette échelle. Cette approche implique que le nombre de pixels associés au bruit domine largement ceux qui seront associés aux structures réelles dans cette image. Un processus itératif de rejet des valeurs extrêmes a donc été mis en place pour limiter l'influence de ces structures sur la détermination de l'écart-type lié au bruit. Pour le cas extrême des images denses, l'utilisateur a toute latitude pour modifier le seuil qui lui sera proposé.

Un mode opératoire pour obtenir une image reconstruite de qualité peut être le suivant. On considèrera d'abord l'option par défaut de mesurer l'écart-type du bruit à partir de tous les pixels des plans après un certain nombre de rejets à trois sigma. L'image reconstruite peut alors être affectée d'un bruit résiduel encore trop important (seuils trop bas) ou ne plus contenir certains objets (seuils trop hauts). Dans ce cas on peut recommencer l'opération en modifiant interactivement la valeur du seuil proposé pour chaque échelle. Pour connaître au mieux celle-ci, la stratégie la plus adéquate consiste à mesurer directement la variance des coefficients sur une zone restreinte du plan en ondelette dépourvu de signal. Il faut au préalable disposer des plans en ondelettes par le biais du programme *trans–wave*, puis utiliser le programme *stat*. D'excellents résultats sont *a priori* obtenus avec comme seuils pour les échelles 1 et suivantes $4\sigma_1$, $3{,}5\sigma_2$, $3\sigma_3$, $3\sigma_4$, $3\sigma_5$ et ainsi de suite. D'autres

stratégies sont possibles (cf. Donoho 1994). On pourra vérifier la qualité des sélections en procédant à une reconstruction globale de l'image à partir des plans en ondelettes seuillés au moyen du programme *rec–image*.

La base d'ondelettes générée par une fonction d'échelle B-spline cubique bidimensionnelle n'est pas orthogonale. Les simulations réalisées sur des images de bruit gaussien d'ecart-type unité ont néanmoins montré que la décroissance théorique d'un facteur deux d'une échelle sur l'autre était pratiquement vérifiée à partir de la quatrième échelle comme rapporté dans le tableau 2.1. La mesure directe sur les plans en ondelettes de la variance liée au bruit est donc conseillée jusqu'à l'échelle numéro trois.

rang i	1	2	3	4	5	6	7	8	9	10
échelle $a = 2^{i-1}$	1	2	4	8	16	32	64	128	256	512
σ_b (i)	0,8910	0,2007	0,0855	0,0412	0,0204	0,0102	0,0051	0,0026	0,0013	0,0006

TAB. 2.1 – Fluctuation rms des coefficients en ondelettes pour un bruit blanc gaussien stationnaire d'écart-type unité. Les valeurs proviennent d'une statistique construite sur mille échantillons avec comme fonction d'échelle un B-spline d'ordre 3 bidimensionnel.

Dans le cas d'un bruit poissonnien, si la transformée en ondelettes est réalisée telle quelle, l'amplitude de la variation des coefficients est directement liée à celle du signal et varie par conséquent d'un pixel à l'autre. Il est donc exclu de pouvoir définir sur cette transformée un seuil unique qui permettrait de détecter les coefficients dits significatifs. Afin de procéder de la sorte la variance doit être stabilisée. Cela est possible si l'image à traiter subit d'abord une transformation de Anscombe qui, pour peu que le nombre d'événements soit supérieur à cinq ou six, transforme sans erreur perceptible le processus de Poisson en un processus gaussien stationnaire d'écart-type unité. L'identification des pixels significatifs dans la transformée de l'image ainsi modifiée peut alors se dérouler selon la méthode décrite pour le cas d'un bruit gaussien. Fluctuations rms et seuils sont dès lors relatifs aux plans en ondelettes de cette seule image et ne sont donc utilisés que pour obtenir la suite des masques identifiant les regions significatives des autres. La valeur des coefficients de ces régions provient de la transformée de l'image initiale. Les supports connexes définissant les domaines significatifs sont en effet identiques sur les deux transformées en ondelettes dans la mesure où la transformation d'Anscombe ne procède qu'à une simple conversion d'intensité.

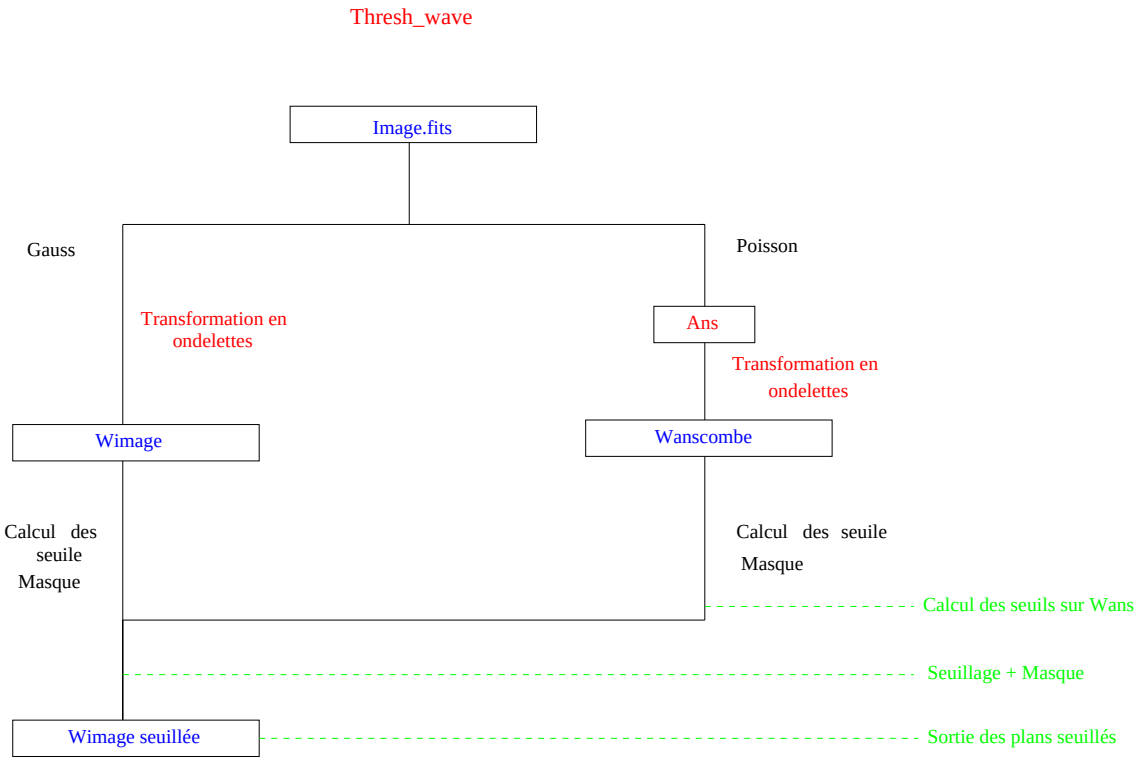


FIG. 2.1 – Seuillage des plans en ondelettes.

– en bleu : les entrées/Ssorties

– en rouge : les procédures

– en vert : les descriptions

La combinaison des bruits de Poisson et de Gauss s'effectue par le biais d'une transformation d'Anscombe généralisée (Murtagh *et al.* 1995).

2.2 Syntaxe du programme thresh–wave

Les différentes étapes du programme *thresh–wave* sont schématisées dans la figure 2.1.

La syntaxe de la procédure *thresh–wave* est la suivante :

14Calcul de la transformée et sélection des coefficients significatifs

Commande :	*thresh–wave* name.fits *-s [-c] [-in] [-m] -b [-gain] [-ron] [-e] [-v] [-ec] [-ans] [*
Entrée :	*name.fits*, l'image à traiter.
Sortie :	*name.ws1.fits* à *name.wsi.fits*, les plans seuillés de la transformée en ondelettes de l'image en entrée.

Description des options :

-s :	nombre d'échelles pour la transformée en ondelettes ;
-c :	coefficient de seuillage des plans (seuil = $c \times \sigma_b$) ;
-in :	nombre de rejets à 3 sigma pour le calcul de σ_b ;
-m :	type de propagation de la variance induite par le bruit sur les coefficients aux différentes échelles (1 : décroissance de moitié, 2 : loi théorique du cas gaussien, 3 : loi théorique du cas gaussien pondéré, [4] : mesure de la variance sur les trois premiers plans puis décroissance de moitié) ;
-b :	nature du bruit ([1] : gaussien, 2 : Poisson pur ou composite) ;
-gain :	gain (ADU/e-) du CCD (cf. loi d'Anscombe généralisée) ;
-ron :	bruit de lecture (e-) du CCD (cf. loi d'Anscombe généralisée) ;
-e :	gestion des effets de bord (1 : continuité, 2 : plongement, [3] : miroir, 4 : repliement) ;
-v :	si *e* égal 2, moyenne de la distribution gaussienne ;
-ec :	si *e* égal 2, écart-type de cette distribution ;
-ans :	sortie des plans après modification de l'image par la loi d'Anscombe (1 pour l'activer, [0] par défaut) ;
-M :	taille (Ko) au-delà de laquelle l'information est copiée temporairement sur disque.

2.3 Exemple d'application

Prenons comme exemple d'image à traiter celle de la figure 2.2. Il s'agit d'un champ de 5,75' × 5,75' extrait de la numérisation d'un cliché de Schmidt ESO-R autour de la galaxie 3840350 du catalogue de Lauberts & Valentijn [ref]. Cette galaxie est partiellement eclipsée par au moins une étoile au premier plan. Ses caractéristiques réelles sont donc difficiles à obtenir ; il faudrait pour cela être capable de séparer proprement l'étoile de la galaxie ... Pour obtenir la transformée en ondelettes sur six échelles de cette image *L384.fits* avec une gestion implicite des effets de bords par miroir, la commande suivante doit être entrée :

> *trans–wave* L384.fits *-s* 6.

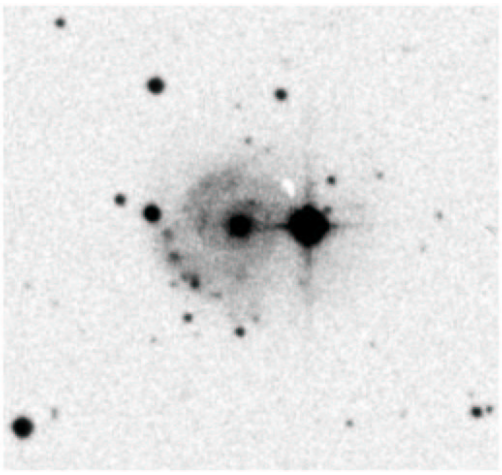


FIG. 2.2 – Image initiale *L384.fits*.

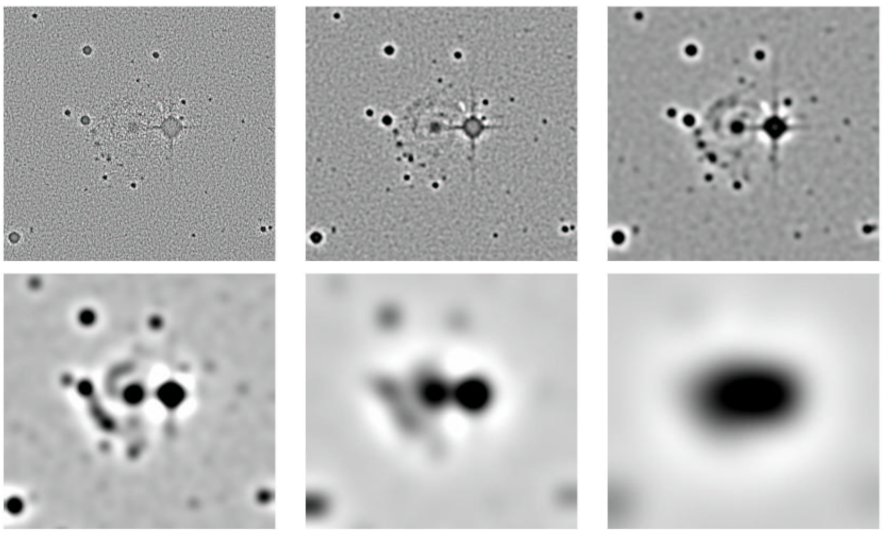


FIG. 2.3 – Les six premières échelles de la transformée en ondelettes de l'image *L384.fits*, depuis $a = 2^0$ en haut à gauche à $a = 2^5$ en bas à droite.

Les plans de la transformée pour les échelles comprises entre $a = 2^0$ et $a = 2^5$, à savoir les images *L384.w1.fits* à *L384.w6.fits* sont représentés sur la figure 2.3. Elle illustre parfaitement l'avantage d'une transformation multiéchelle des données : les détails de différentes tailles apparaissent aux échelles qui leur correspondent. On voit ainsi très bien comment des objets superposés sont perçus comme une structure unique à grande échelle pour s'individualiser progressivement à mesure que des échelles plus petites sont explorées.
Les coefficients de la transformée de l'image *L384.fits* n'ont visiblement pas tous la même valeur ; les plus petits en valeur absolue sont visiblement associés au fond de ciel tandis que les plus grands sont engendrés par les structures présentes dans l'image de départ. En regard de l'amplitude des variations des coefficients du fond, il est clair que ceux reliés aux structures ont des valeurs telles qu'il est très peu probable de les rencontrer par simple fluctuation du fond. On peut donc facilement isoler ces coefficients à la signification statistique élevée en les comparant à l'écart-type des coefficients liés au fond. Pour effectuer cette sélection sur les six premiers plans de la transformée en ondelettes de l'image *L384.fits*, il suffit par exemple de lancer la commande :

> *thresh–wave* L384.fits *-s* 6 *-in* 5.

Dans ce cas précis, l'écart-type sur les coefficients des trois premières échelles est obtenu après 5 itérations où sont éliminés les pixels dont la valeur s'écarte de plus de trois fois l'estimation. La valeur finale de cet écart-type est espérée être une mesure correcte de la fluctuation sur les coefficients induite par le bruit sur l'image initiale. Cet espoir est certainement fondé dans notre exemple où, pour les trois premiers plans, le nombre de pixels du fond est sensiblement plus grand que celui des objets. Pour les plans suivants cela n'est plus le cas et la variance du "bruit" à ces échelles supérieures est donc prise comme égale à la moitié de la valeur de l'échelle précédente comme attendue par la théorie. Le tableau 2.2 rassemble les valeurs obtenues pour les fluctuations rms. Il donne également les valeurs limites proposées par défaut au-delà desquelles un coefficient sera jugé comme significatif, à savoir un seuil de trois fois l'écart-type. Les plans seuillés selon cette série de valeurs sont présentés figure 2.4.

On peut constater qu'un nombre relativement important de pixels de fond ont été retenus comme pertinents sur les deux premiers plans. Cela est somme toute normal. Avec un seuil de 3σ près de quatre-vingt-dix fausses alarmes sont attendues sur le premier plan pour une image de 65536 pixels. On peut donc décider d'augmenter les seuils pour ces deux plans en prenant des valeurs correspondant par exemple à respectivement 4σ et $3{,}5\sigma$. La nouvelle série de

Echelle	σ initial	σ après 5 rejets	Seuil : $3 \times \sigma$
1 : $a = 2^0 = 1$	41,461	37,736	113,208
2 : $a = 2^1 = 2$	37,594	18,309	54,927
3 : $a = 2^2 = 4$	51,177	12,179	36,537
4 : $a = 2^3 = 8$			18,268
5 : $a = 2^4 = 16$			9,134
6 : $a = 2^5 = 32$			4,567

TAB. 2.2 – Seuils adoptés par défaut pour les 6 premiers plans de la transformée en ondelettes de l'image *L384.fits*.

Echelle	σ initial	σ après 5 rejets	Seuils ajustés
1 : $a = 2^0 = 1$	41,461	37,736	150,944
2 : $a = 2^1 = 2$	37,594	18,309	64,081
3 : $a = 2^2 = 4$	51,177	12,179	36,537
4 : $a = 2^3 = 8$			18,268
5 : $a = 2^4 = 16$			9,134
6 : $a = 2^5 = 32$			4,567

TAB. 2.3 – Seuils choisis pour les six premiers plans de la transformée en ondelettes de l'image *L384.fits*. Ceux des deux premières échelles correspondent à respectivement 4 et 3,5 fois l'écart-type mesuré, les suivants à 3 fois l'écart-type.

seuils qui en découle est consignée dans le tableau 2.3. Seule la sélection sur les deux premiers plans en ondelettes s'en trouve affectée ; leur nouveau seuillage est présentée figure 2.5.

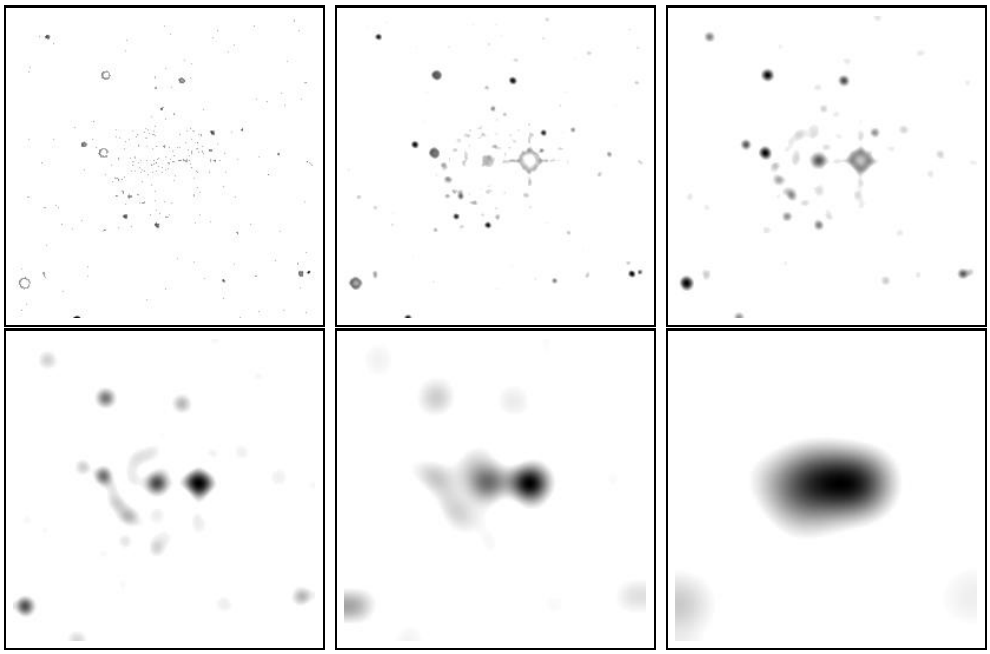


FIG. 2.4 – Régions significatives à 3σ sur les six premiers plans de la transformée en ondelettes de l'image *L384.fits*.

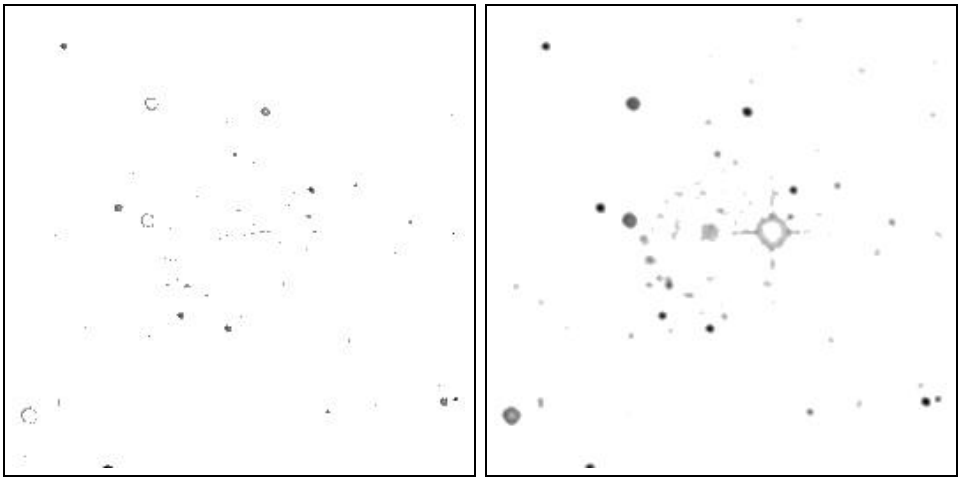


FIG. 2.5 – Régions significatives sur les deux premiers plans en ondelettes de l'image *L384.fits* après ajustement des seuils de sélection.

Chapitre 3

Segmentation des régions significatives et graphes interéchelles

3.1 Segmentation et sous-segmentation des plans en ondelettes seuillés

Pour chacun des plans, les pixels significatifs sont regroupés en domaines connexes (voisinage à quatre pixels) et se voient affectés d'une même étiquette au sein d'un domaine donné. Les domaines sont numérotés de façon croissante en fonction de la position (colonne, ligne) de leur premier pixel. L'étiquette nulle est réservée au domaine des pixels non significatifs.

A la fin de cette étape de segmentation, chaque domaine ainsi identifié peut contenir plusieurs maxima locaux qui traduisent l'existence de composantes à plus petite échelle (cf. la linéarité de l'opérateur "transformation en ondelettes"). La définition des sous-domaines reliés à chaque maximum est possible par le biais d'une sous-segmentation. Ce raffinement est un gage de qualité lorsqu'est introduite la notion de structure individuelle à mesurer séparément. L'algorithme mis actuellement en place pour réaliser cette sous-segmentation est basé sur un simple critère de distance euclidienne ; chaque pixel est rattaché au maximum local dont il est le plus proche.

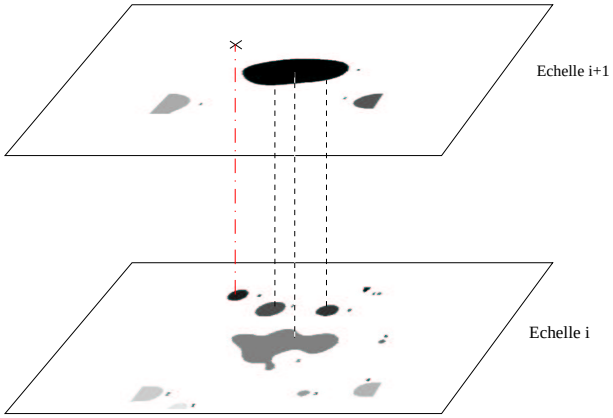


FIG. 3.1 – Un lien interéchelle entre deux domaines est établi si la position du coefficient le plus grand du domaine d'échelle inférieure se projette à l'intérieur du domaine de l'échelle supérieure. Dans cet exemple, ce n'est pas le cas du domaine de l'échelle i repéré par la liaison alternant points et tirets.

3.2 Etablissement des connections interéchelles

Les domaines et sous-domaines de chaque échelle ayant été définis, ils sont mis en relation selon le critère qui suit. Examinant les domaines depuis l'échelle la plus petite vers les plus grandes, un lien interéchelle est établi entre deux domaines si la position du maximum du domaine testé se projette à l'intérieur d'un domaine de l'échelle immédiatement supérieure (cf. figure 3.1). Ce faisant, deux règles sont respectées :

- un domaine de l'échelle n ne peut être lié qu'à un seul domaine de l'échelle $n+1$;
- un domaine ne doit être lié qu'à un domaine de l'échelle immédiatement supérieure dans la mesure où la hiérarchie de structures d'un objet physique ne peut être interrompue.

L'établissement de ces liens interéchelles débouche sur un ensemble de graphes décrivant chacun la hiérarchie locale des domaines significatifs telle que mise en évidence par la transformation en ondelettes. Au-delà du seul codage de cette arborescence, chaque graphe ou *arbre* mémorise également la valeur des pixels de tous les domaines concernés. Les domaines n'ayant pas de lien avec un domaine sis à l'échelle supérieure définissent ce qu'il est convenu d'appeler la racine de l'*arbre*. Elle sert à numéroter les arbres de façon séquentielle avant leur sauvegarde au format FITS. Remarquons qu'il ne s'agit pas là

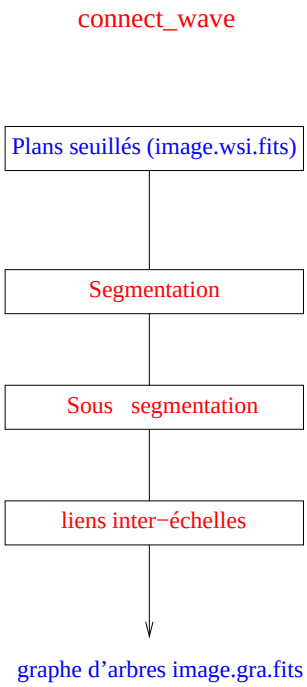


FIG. 3.2 – Les différentes étapes conduisant aux connections interéchelles (en bleu, les entrées/sorties ; en rouge, les étapes algorithmiques).

d'une image ; l'entête du fichier comprend différentes informations sur le résultat de la segmentation et le nombre d'arbres trouvés utiles pour les étapes ultérieures du traitement.

3.3 Syntaxe de la procédure connect–wave

Les différentes étapes du programme *connect–wave* sont indiquées sur la figure 3.2.

La syntaxe de la procédure *connect–wave* est la suivante :

Commande :	*connect–wave* name.fits *-nbre [-S] [-seg] [-sseg] [-M]*
Entrée :	*name.fits*, le nom de l'image afin de pouvoir générer la liste des noms des plans en ondelettes seuillés de sa transformée en ondelettes.
Sortie :	*name.gra.fits*, un fichier contenant l'ensemble des graphes décrivant l'arborescence des liens interéchelles avec la valeur des pixels des domaines ainsi connectés.

Description des options :

-nbre :	nombre de plans à connecter à partir du plan 1 ;
-S :	option de sous-segmentation (1 pour l'activer, [0] par défaut) ;
-seg :	sauvegarde des plans segmentés (1 pour l'activer, [0]) ;
-sseg :	sauvegarde des plans sous-segmentés (1 pour l'activer, [0]) ;
-M :	taille (Ko) au-delà de laquelle l'information est temporairement copiée sur disque.

3.4 Exemple d'applications

La segmentation des six premiers plans de la transformée en ondelettes de l'image *L384.fits* avec sauvegarde des plans s'obtient par la commande :

> *connect–wave* L384.fits *-S* 0 *-seg* 1 *-nbre* 6

La segmentation et sous-segmentation de cette même transformée sans sauvegarde des plans sera le fait de la commande :

> *connect–wave* L384.fits *-S* 1 *-sseg* 0 *-nbre* 6

La figure 3.3 présente le résultat de cette opération de segmentation tandis que la figure 3.4 montre celui obtenu en effectuant également une sous-segmentation. Un exemple de codage de l'arborescence des structures mémorisée dans le fichier *L384.gra.fits* est fourni par la figure .... obtenue au moyen du programme *dessine–arbre* (cf. annexe).

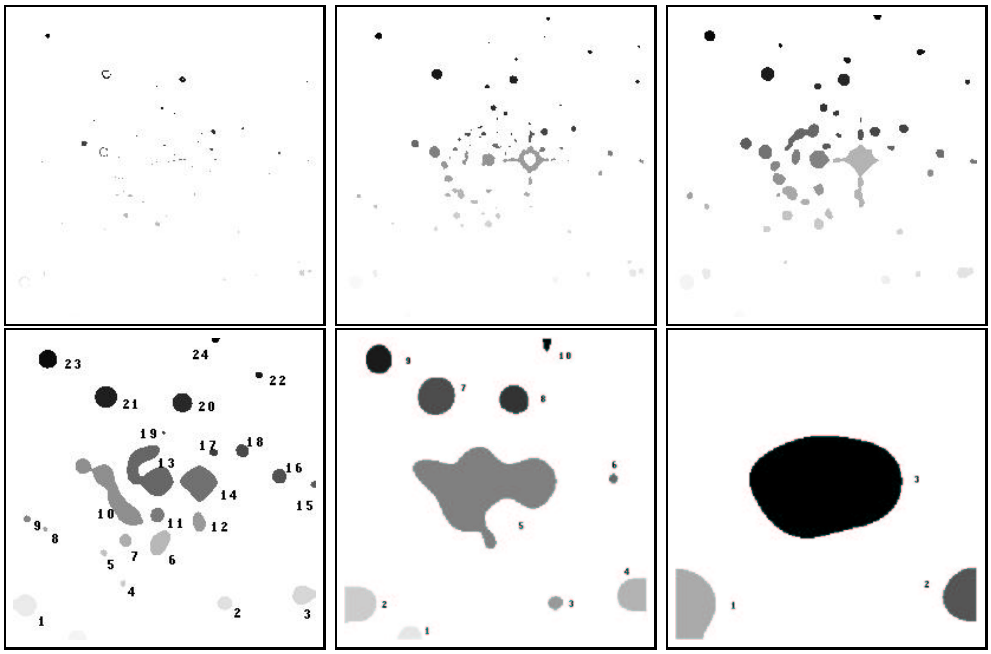


FIG. 3.3 – Segmentation des régions significatives de la transformée en ondelettes de l'image *L384.fits*. Chaque domaine connexe se voit affecté une même étiquette visualisée sous l'aspect d'un niveau de gris particulier. Sa valeur n'est affichée que pour les domaines des plans 4 à 6.

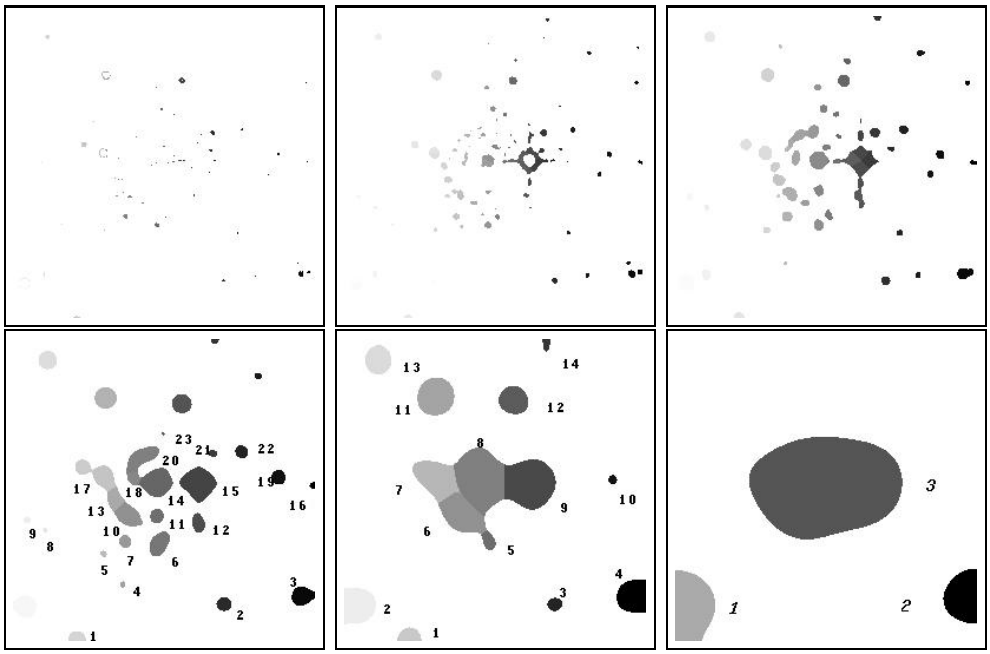


FIG. 3.4 – Résultat d'une segmentation suivie d'une sous-segmentation pour les régions significatives de la transformée en ondelettes de l'image *L384.fits*.

Chapitre 4

Identification de structures

4.1 Etapes du programme obj–wave

Examinant l'ensemble des graphes créés par le programme *connect-wave* dans l'ordre de leur numérotation, le programme *obj-wave* cherche à détecter parmi les domaines significatifs de chaque arbre ceux que l'on peut considérer comme représentatifs des structures présentes dans l'image de départ. Rappelons en effet qu'en l'absence de bruit chacune d'entre elles se manifeste par des coefficients significatifs à toutes les échelles de la transformée et qu'il faut donc pour les caractériser par une taille typique singulariser à l'échelle correspondante un domaine particulier. Cela sera le fait d'une réponse maximale. Parcourant chaque arbre depuis sa racine (domaine situé à l'échelle la plus grande) vers ses extrémités (les domaines d'échelle la plus petite pour chacune des branches), on considère déceler une structure d'échelle caractéristique a lorsque le coefficient maximum du domaine testé à l'échelle a a une valeur supérieure à celles des maxima des domaines qu'on lui a associés aux échelles supérieures et inférieures. Récursif sur tous les fils de la généalogie, ce principe de base demande toutefois quelques ajustements pour être opérationnel. De fait, pour le domaine associé à l'échelle supérieure, l'influence d'éléments à cette même échelle ou les valeurs induites par des éléments de plus petite taille mais très intenses peuvent faire que la valeur et l'emplacement de son maximum n'a plus beaucoup de lien avec la contrepartie à cette échelle du domaine examiné. Pour contrecarrer ces effets biaisant le test de comparaison, le maximum considéré sera donc celui de la région définie par la projection sur ce domaine du domaine en cours d'évaluation. L'examen de l'échelle inférieure peut de la même façon être perturbé par une trop grande proximité de plusieurs éléments se manifestant par une série de maxima au sein du domaine regardé. Considérant qu'une sous-segmentation pallie cet

inconvénient, il a été décidé de ne regarder dans la filiation que le domaine le plus voisin de la position de celui évalué.

4.2 Syntaxe de la procédure *obj–wave*

Les différentes étapes du programme *obj–wave* sont schématisées figure 4.1.

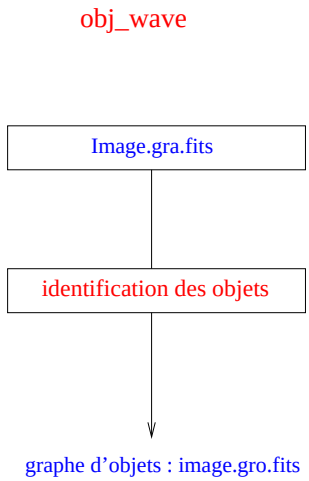


FIG. 4.1 – Identification des structures dans l'image directe (en bleu, les entrées/sorties ; en rouge, les étapes méthodologiques).

La syntaxe de la procédure *obj–wave* est la suivante :

Commande :	*obj–wave* name.fits *[-D] [-M]*
Entrée :	*name.fits*
Sortie :	*name.gro.fits*, un fichier contenant la restriction aux objets des graphes interéchelles auxquels ont été ajoutées les informations les concernant (ex. : position, échelle, amplitude).

Description des options :

-D :	comparaison des coefficients en amplitude ([0]) ou en énergie (1) ;
-M :	taille en Ko à partir de laquelle les plans sont temporairement copiés sur disque.

Remarque : les comparaisons peuvent porter sur l'amplitude w des coefficients ou sur leur énergie $E = w \times 2^i$ avec i le rang de l'échelle considérée.

4.3 Exemple

L'identification de structures à partir de la *forêt d'arbres* se fait par la commande :

> *obj–wave* image.gra.fits,

avec comme résultat le graphe des structures *image.gro.fits*. On peut alors décider de reconstruire l'image à l'aide de ce graphe par la commande (cf. chapitre VI) :

> *rec–obj* L384.gro.fits.

Des outils de visualisations du graphe de structures sont décrits en annexe. Pour obtenir par exemple la carte des structures identifiées, on peut ainsi faire appel à la procédure *carte–obj* en saisissant :

> *carte–obj* name.fits -*num*.

La carte au format postcript qui résulte de cette exécution se nomme *name–carte.ps*; elle est représentée figure 7.25.

Chapitre 5

Reconstructions

5.1 La méthode du gradient conjugué

A partir du jeu restreint de coefficients en ondelettes mémorisés soit dans les plans seuillés soit dans la forêt d'arbres issue de la détermination des connections interéchelles et de l'identification des structures qui peut suivre, il s'agit de reconstruire une image dans l'espace direct. Les coefficients ayant été sélectionnés vis-à-vis de l'impact du bruit sur leurs valeurs, cette image finale sera débruitée.

Appelons I l'image de départ, A l'opérateur réalisant la transformation en ondelettes, M celui associé à la sélection des coefficients significatifs, W la transformée et W_s sa restriction aux coefficients dits significatifs. Nous avons : $W = A(I)$ et $W_s = M(W)$. L'objectif est donc d'obtenir une image $\tilde{I}$ satisfaisant $W_s = (MoA)(\tilde{I})$ ou encore $\tilde{A}(W_s) = (\tilde{A}oMoA)(\tilde{I})$ avec $\tilde{A}$ l'opérateur adjoint de (MoA). Il s'agit là d'un problème inverse admettant en général une infinité de solutions. Une d'entre elles doit être sélectionnée par l'entremise d'un critère de régularisation. Cherchant à minimiser à chaque étape la fonctionnelle $E(\tilde{I}) = ||\tilde{A}(W_s) - (\tilde{A}oMoA)(\tilde{I})||^2$ où $||\ ||$ dénote la distance euclidienne, l'algorithme mis en place pour trouver cette solution s'appuie sur une méthode de gradient conjugué dans laquelle, pour accélérer la convergence, le calcul de la direction de descente à une itération donnée fait intervenir les gradients trouvés aux itérations précédentes [ref ...]. Une contrainte de positivité est instaurée afin d'aboutir à une image sans valeurs négatives.

5.2 Reconstruction à partir des plans seuillés

5.2.1 Le programme *rec–image*

Le programme *rec–image* use de l'algorithme de gradient conjugué évoqué dans le paragraphe précédent pour construire une image débruitée à partir des coefficients en ondelettes sélectionnés par le programme *thresh–wave*. Ne faisant appel ni aux connections interéchelles ni à une identification préalable des structures, l'opération a l'avantage d'être assez rapide et permet de vérifier à moindre coût la validité des seuils adoptés. En contrepartie l'évaluation de la fonctionnelle porte sur l'ensemble des coefficients et, si la qualité globale de la reconstruction est assurée, la pertinence locale de la solution sera donc d'autant meilleure que les propriétés statistiques de l'image sont identiques de place en place. Soulignons par ailleurs que la nécessité de disposer de toute la transformée peut exiger des ressources disque importantes.

5.2.2 Syntaxe de la procédure rec–image

Les différentes étapes de la procédure *rec–image* sont indiquées figure 5.1.

La syntaxe de la procedure *rec–image* est la suivante :

Commande :	*rec–image* name.fits *[-ir] -Echmin -Echmax [-gef] [-v] [-sig] [-e] [-M]*
Entrée :	*name.fits*, l'image traitée afin de générer les différents noms. ;
Sortie :	*name–recimage.fits*, l'image reconstruite.

Description des options :

-ir :	nombre maximal d'itérations pour atteindre la convergence ;
-Echmin :	première résolution à considérer (commence à 1) ;
-Echmax :	dernière échelle à intégrer dans la reconstruction ;
-gef :	gestion des effets de bord (1 : continuité, 2 : plongement, [3] : miroir, 4 : repliement) ;
-v :	si *gef* égal 2, moyenne de la distribution gaussienne ;
-sig :	si *gef* égal 2, écart-type de cette distribution ;
-e :	critère d'arrêt sur la valeur de la fonctionnelle [0.0] ;
-M :	taille (Ko) à partir de laquelle les plans sont temporairement copiés sur disque.

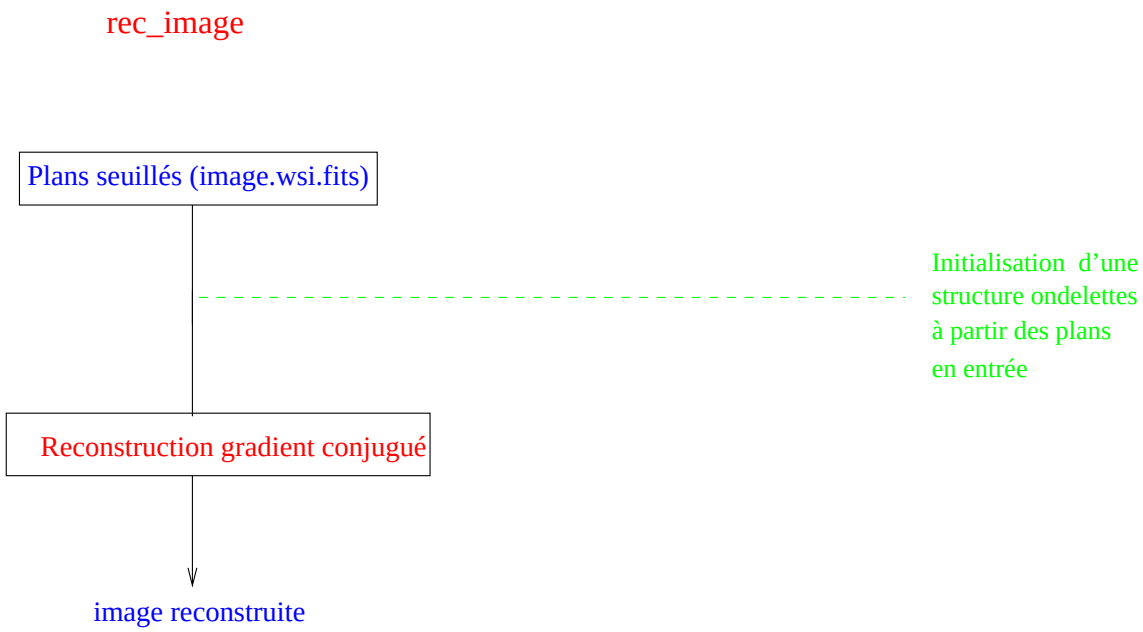


FIG. 5.1 – Reconstruction d'une image débruitée.

- en bleu : les entrées/sorties
- en rouge : les procédures
- en vert : les descriptions

5.2.3 Exemple

Pour obtenir une version débruitée de l'image qui nous sert de test à l'aide des détails des échelles 1 à 6 et de cinq itérations de l'algorithme de gradient conjugué, la commande suivante doit être tapée :

> *rec–image* L384.fits *-ir* 5 *-Echmin* 1 *-Echmax* 6.

L'image en sortie, *L384–recimage.fits*, est donnée figure 5.3.

5.3 Reconstructions à partir d'arbres

Un graphe de connection interéchelles ou arbre permet de regrouper tous les domaines significatifs situés dans une même région de l'image moyennant les conditions d'inclusion décrites à la section 2.2. Cela étant réalisé, l'option d'une reconstruction par arbres permet de construire l'image finale comme la superposition de toutes les imagettes obtenues à partir des séries de domaines mis ainsi en relation. L'avantage de cette approche est d'avoir une reconstruction optimisée pour chacun des ensembles de coefficients significatifs et donc une meilleure représentation des structures présentes dans l'image finale.
Tel qu'il est programmé l'algorithme de gradient conjugué sous-jacent aux reconstructions ne considère que la transformée d'une image par un algorithme à trous, c'est-à-dire une série de plans de même taille contenant chacun les coefficients en ondelettes d'une même échelle. Les domaines mémorisés dans chaque arbre ayant une toute autre organisation, l'ensemble de ces pixels est donc réorganisé selon la structure attendue avant de lancer l'algorithme. Cela contribue à accroître les temps de calcul.

5.3.1 Syntaxe de la procédure rec–obj

Les différentes étapes de la procédure *rec–obj* sont schématisées figure 5.2.

La syntaxe de la procédure *rec–obj* est la suivante :

Commande :	*rec–obj* name.fits *[-A] [-ef] [-v] [-ec] [-Brec] [-Erec] -ir -e [-L] [-F] [-cat] [-M]*
Entrée :	*name.fits*, l'image traitée afin de générer les différents noms ;
Sorties :	*name–recobj.fits* ou *name–recarbre.fits*, l'image reconstruite obtenue avec ou sans identification des structures ;
	name.g.param, les paramètres des structures identifiées.

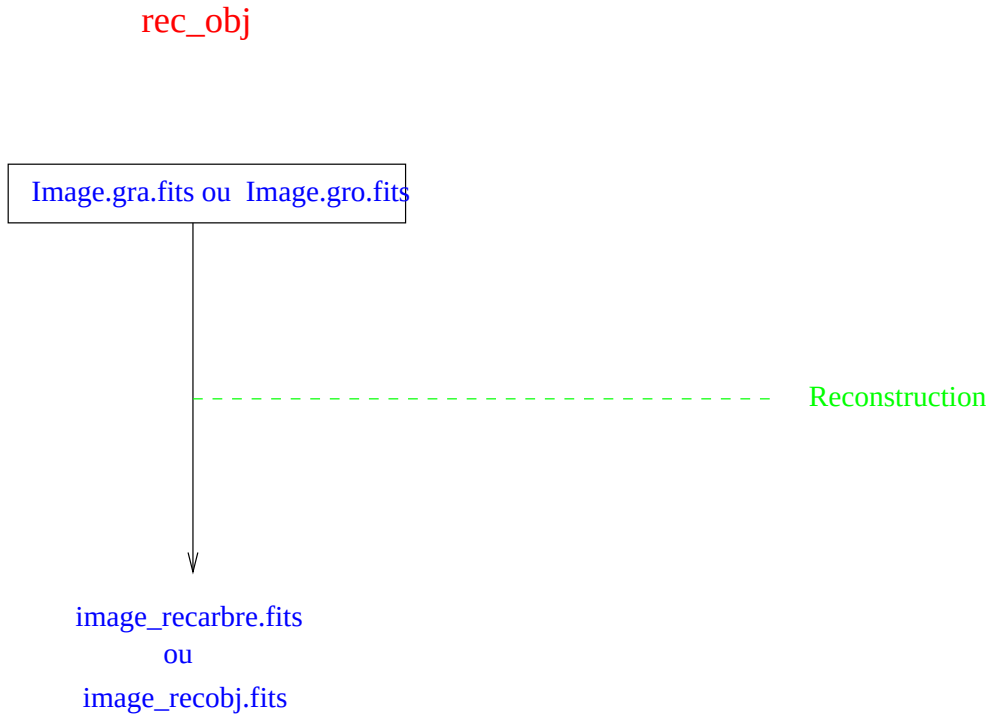


FIG. 5.2 – Reconstruction avec ou sans identification

- en bleu : les entrées/sorties
- en rouge : les procédures

Description des options

-A :	option de reconstruction sans identification des structures ;
-ef :	gestion des effets de bord (1 : continuité, 2 : plongement [3] : miroir, 4 : repliement) ;
-v :	si *ef* égal 2, moyenne de la distribution gaussienne ;
-ec :	si *ef* égal 2, écart-type de cette distribution ;
-Brec :	échelle des plus petites structures reconstruites ;
-Erec :	échelle des plus grandes structures reconstruites ;
-ir :	nombre maximal d'itérations pour atteindre la convergence ;
-e :	critère d'arrêt sur la valeur de la fonctionnelle [0.0] ;
-L :	fichier optionnel avec les attributs de reconstruction des différentes structures et sous-structures identifiées ;
noindent*-F* :	option pour produire l'image individuelle de chaque structure ;
-cat :	Option pour créer un catalogue (vérifier l'initialisation du fichier *mvm–rec.conf*) ;
-M :	taille (Ko) à partir de laquelle les plans sont temporairement copiés sur disque.

5.3.2 Exemples

La reconstruction sans identification donnée figure 5.4 a été obtenue par la commande :

> *rec–obj L384.fits -A -ir* 5.

La reconstruction avec identification aurait été le fait de la commande :

> *rec–obj L384.fits -ir* 5.

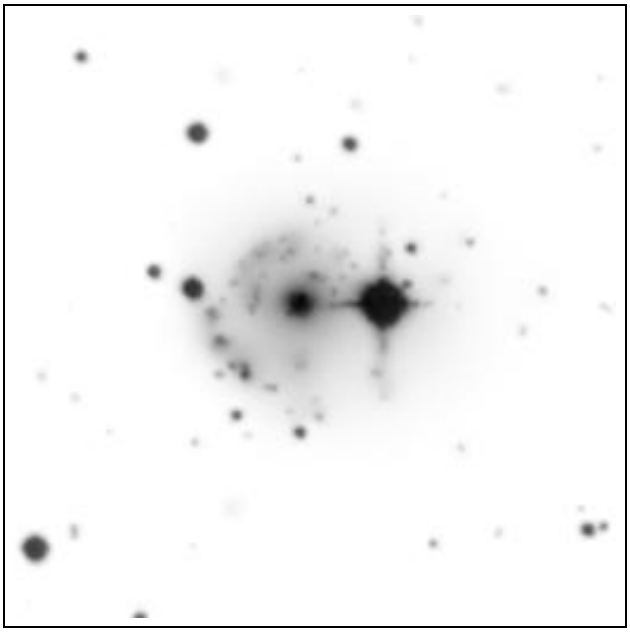


FIG. 5.3 – Image obtenue par le biais d'une reconstruction à l'aide des plans seuillés.

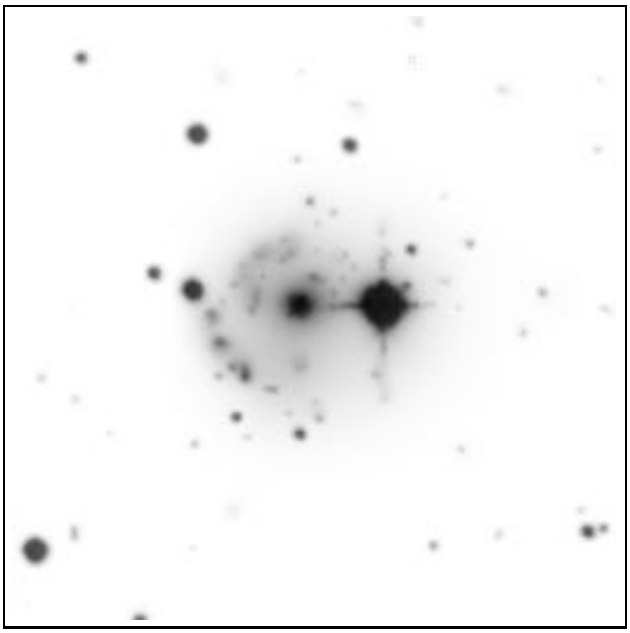


FIG. 5.4 – Image obtenue par le biais d'une reconstruction par arbres sans identification des structures.

Chapitre 6

Le programme *mvm–rec*

Il s'agit là d'un programme global offrant à l'utilisateur la possibilité d'exécuter en une seule commande tout ou partie des opérations décrites dans les chapitres précédents, les résultats étant sauvegardés dans un répertoire *image–date1* créé automatiquement.
La syntaxe de ce programme est la suivante :

Commande :	*mvm–rec image.fits image.conf*;
Entrées :	– *image.fits*, l'image à traiter ;
	– *image.conf*, les paramètres configurant l'exécution ;
Sorties :	– *image.log*, un mouchard de l'exécution ;
	– l'image reconstuite selon le mode sélectionné ;
	– le catalogue optionnel des structures identifiées.

6.1 Fichier de configuration

Le fichier de configuration *image.conf* contient tous les paramètres nécessaires à l'exécution du programme depuis le calcul de la transformée en ondelettes jusqu'à l'obtention des paramètres du catalogue. Le voici *in extenso* :

```
*********************************************************************
```

Fichier de configuration de MVM Version 1.0a

type de reconstruction

TYPE–REC : > 1 pour Reconstruction par plans seuillés
> 2 pour Reconstruction par arbres
> 3 pour Reconstruction par structures

transformation en ondelettes et seuillage automatique :

THRESH : *-s* 5 > *Nombre de plans*
THRESH : *-b* 1 > *Type de bruit : 1 pour gaussien, 2 pour poisson*
THRESH : *-in* 3 > *Nombre d'itérations pour le calcul du bruit*
THRESH : *-c* 3.0 > *Coefficient de seuillage des plans ;seuillage :*
> $c \times$ *sigma (bruit des plans)*

#cas de bruit de Poisson

THRESH : *-gain* 1.0 > *(ADU/e^-) pour la transformée d'Anscombe généralisée*
THRESH : *-ron* 0.0 > *(e^-) pour la transformée d'Anscombe généralisée*

#Gestion des effets de bords

THRESH : *-e* 3 > 1 : continuité
> 2 : plongement
> 3 : miroir
> 4 : repliement

#valeurs par défaut

THRESH : *-v* 0 > valeur de plongement si $e = 2$
THRESH : *-ec* 0.0 > écart-type du bruit gaussien si $e = 2$
THRESH : *-M* 1024 > taille (Ko) à partir de laquelle les plans sont temporairement copies sur disque

paramètres de la reconstruction par plans

REC–GLOB :	*-ir*	5	> nombre d'itérations pour la reconstruction
REC–GLOB :	*-Echmin*	1	> échelle à partir de laquelle on veut reconstruire
REC–GLOB :	*-Echmax*	5	> échelle max de la reconstruction
REC–GLOB :	*-e*	0.001	> Ecart maximal entre la reconstruction choisie et la première

paramètres pour la reconstruction par arbres

REC–AV–IDENT :	*-S*	0	> mettre à 1 pour opter pour une sous-segmentation
REC–AV–IDENT :	*-seg*	0	> option de sauvegarde des plans segmentés
REC–AV–IDENT :	*-sseg*	0	> option de sauvegarde des plans sous-segmentés
REC–AV–IDENT :	*-nbre*	5	> nombre de plans à connecter à partir de 1

paramètres pour l'identification des structures

IDENT :	*-D*	0	> paramètre pour l'identification par défaut en amplitude, 1 pour énergie

paramètres pour la reconstruction avec identification des structures

REC–OBJ :	*[-L]*	name.lst	> nom d'une liste d'objets à reconstruire créée avec make–liste
REC–OBJ :	*-cat*	0	> option pour sortir un catalogue des objets identifiés
REC–OBJ :	*-Brec*	1	> plan à partir duquel les objets sont reconstruits (de 1 à 20)
REC–OBJ :	*-Erec*	20	> plan limite pour la reconstruction des objets (de 1 à 20)
REC–OBJ :	*-ir*	10	> nombre d'itérations maximales pour la reconstruction
REC–OBJ :	*-e*	0.001	> écart maximal entre la reconstruction choisie et la première
REC–OBJ :	*[-F]*		> option pour reconstruire les objets sur des images séparées

paramètres pour la sortie de catalogue

CATALOGUE		***Mot clef indispensable pour la sortie de catalogue***
Carte =	*F*	> option d'une "Carte des variations du fond"
Nom1 =		> carte étendue du fond de ciel
NBl =	0	> dimension Y
NBp =	0	> dimension X
iWave1 =	0	> échelle de détection
iWave2 =	0	> numéro de l'objet
Nom2 =	*name.fits*	> image de travail au format FITS
Nl =	10	> dimension Y de l'imagette
Npl =	10	> dimension X de l'imagette
YOrg =		> coordonnée Y du pixel (1,1) de l'imagette
XOrg =		> coordonnée X du pixel (1,1) de l'imagette
Seuil =	0	> seuil au-dessus du fond de ciel (adu)
SeuilM =	16000	> seuil de saturation sur l'imagette
Nom3 =	*imagette.fields*	> fichier (provisoire) regroupant les moments de chaque domaine
Nom4 =	*imagette.g.ellfit*	> fichier (provisoire) des domaines ajustés par une ellipse
iGeo =	1	> moments géométriques(1) ou barycentriques(0)
DInt =	3.0	> distance d'intégration : DInt × Ax (unité : longueur du demi-grand axe)
NIter =	3	> NIter-1 rejets a rNSig sigma
dNSig =	3.0	> rNSig-sigma cliping
iResol =	1	> sur-échantillonnage iResol×iResol des pixels
Nom5 =	*imagette.g.prof*	> fichier (provisoire) avec le profil photométrique de l'objet regroupant sur 3 lignes le profil, l'écart-type des valeurs et la surface des couronnes elliptiques
Gain =	1.0	> valeur du gain (e^-/adu)
RON =	0.	> valeur du bruit de lecture (e^-)
PixSize =	1.	> dimension du pixel en seconde d'arc
rTime =	1.	> durée d'intégration (s)
HWHM =	1.	> demi-largeur de la gaussienne du seeing (pixel)
ZCal0 =	26.	> calibration des magnitudes (adu/s/pixel)
Nom6 =	*imagette.g.bright*	> fichier (provisoire) avec les courbes de croissance calculées, à savoir la brillance locale, son écart à la brillance moyenne et la magnitude intégrée
Nom7 =	*name.g.param*	> fichier des paramètres mesurés
iType =	2	> interpolation linéaire (1) ou B3-Spline (2) des valeurs du profil photométrique

Les paramètres associés à la transformation en ondelettes et à la sélection des coefficients sont les mêmes que ceux utilisés pour exécuter le programme *thresh–wave* décrit au chapitre 1.
Le paramètre ***TYPE–REC*** détermine quel type de reconstruction est choisi, à savoir : i) une reconstruction par plans seuillés (cf. section 4.3.1) ; ii) une reconstruction par arbres sans identification des structures (cf. section 4.4) ; une reconstruction par arbres avec identification des structures (cf. section 4.4). Si le type de reconstruction choisi est une reconstruction par plans seuillés, les paramètres qui interviennent sont ceux précédés du mot clef ***REC–GLOB***. Si une reconstruction par arbres a été choisie, les paramètres qui interviennent sont ceux précédés des mots clefs ***REC–AV–IDENT*** et ***REC–OBJ***; Ils interviennent dans les programmes *connect–wave* détaillé au chapitre 2 et *rec–obj –A* présenté en section 4.3. Si l'option d'identifier les structures a été retenue, les paramètres qui interviennent sont ceux précédés des mots clefs ***IDENT*** et ***REC–OBJ***. ils sont associés aux programmes *identif–obj* et *rec–obj* décrits respectivement chapitres 3 et 4.

6.2 Paramétrisation des objets

Le modèle de vision multiéchelle employé offre plusieurs avantages. Il permet d'améliorer l'image de l'objet à étudier en y supprimant le bruit sans pour autant effacer les détails fins à conserver et offre la possibilité de le corriger des sources proches le parasitant. Rappelons par ailleurs que l'image restaurée l'est obligatoirement à une composante uniforme près et qu'en conséquence l'estimation du fond local et sa soustraction à chacun des pixels n'y sont plus à faire. Cette image peut donc être analysée avec le plus grand profit pour caractériser l'objet dans tous ses aspects. Les grands axes de la méthodologie mise en oeuvre pour cela peuvent être trouvé dans plusieurs articles de la littérature (ex. : Slezak *et al.* 1988; Bertin & Arnouts 1996). Résumons en les étapes.
La stratégie adoptée pour acquérir les mesures nécessaires respose sur une classique segmentation d'image où tout objet est posé comme un ensemble connexe de pixels avec la même propriété, en l'occurence une intensité non nulle (cf. fond inexistant). A mesure de l'étiquetage de ces pixels, la superficie au-dessus du seuil, le flux englobé, la valeur du pixel le plus brillant et les deux premiers moments géométriques et barycentriques sont calculés pour chaque domaine (un seul *a priori*. Ces quantités sont ensuite ramenées par réduction à des grandeurs physiques simples (ex. : position) et à des paramètres morphologiques décrivant la forme globale du domaine (ex. :elongation, angle de position). L'objet examiné se voit alors modélisé par une structure à symétrie

de révolution dont les variations centre-bord d'intensité sont codées par un profil radial. Il est construit en moyennant avec réjection itérative l'intensité des pixels au sein de couronnes homothétiques à l'ellipse de forme associée à l'objet ; un raffinement de maillage est possible. Evalué jusqu'à trois, voir cinq fois le demi-grand axe, ce profil est mis à profit pour paramétrer la distribution de brillance au sein de l'objet et estimer sa luminosité, cela une fois saisies les informations de calibration requises. Outre la brillance centrale et moyenne de l'objet sont ainsi mesurées trois magnitudes différentes (isophotale (26 mag/arcsec^2), de Kron, asymptotique de Petrossian) avec leur incertitude respective, plusieurs rayons (isophotal, de Kron, de Petrossian, effectif, équivalents), deux indices de concentration (rapports du rayon effectif au rayon isophotal et à celui regroupant 80% du flux total), deux gradients (au rayon effectif et entre le centre de l'objet et la brillance au demi-grand axe) ainsi qu'un écart global à un profil gaussien d'écart-type défini. Ce jeu assez complet de paramètres a été mis en place afin de pouvoir séparer si besoin les objets d'après leurs vecteurs de forme entre sources ponctuelles et diffuses.

Les paramètres ainsi mémorés dans le fichier de paramètres de sortie sont les suivants :

iRank :	*échelle d'analyse*;
Numero :	*numéro d'ordre du domaine*;
CoordX,CoordY :	*coordonnées cartésiennes sur l'image*;
Aire :	*superficie du domaine (arcsec^2)*;
BMax :	*brillance maximale du domaine (mag/arcsec^2)*;
NSatur :	*nombre de pixels du domaine au-dessus du seuil de saturation*;
AxA,BxA :	*demi-grand et demi-petit axes du domaines (arcsec)*;
Angle :	*angle de position du domaine (degrés, sens trigo.)*;
rMagP,DrMagP :	*magnitude asymptotique de Petrossian et sa précision*;
RPetroA :	*rayon de Petrossian (arcsec)*;
BDPetro :	*brillance sur le profil à la distance de Petrossian (deux fois le rayon de Petrossian)*;
BFond :	*brillance du fond local*;
BMoy :	*brillance moyenne de l'objet*;
BPetro :	*brillance sur le profil au rayon de Petrossian*;
rMagI,DrMagI :	*magnitude isophotale et son incertitude*;
RIsoA,BIso :	*rayon isophotal (arcsec) et isophote adoptée*;
rMagK,DrMagK :	*magnitude d'ouverture de Kron et sa précision*;
RKronOA :	*rayon équivalent (arcsec)*;
BDKron :	*brillance sur le profil à l'ouverture de Kron (2,5 fois le rayon équivalent)*;
RKron1A,RKron2A :	*rayons effectifs (arcsec)*;
C80,CIso :	*paramètres de concentration du profil*;
BCentre :	*brillance centrale*;
B50,R50A :	*brillance sur le profil au rayon effectif et valeur de celui-ci (arcsec)*;
G50,GAx :	*gradients de brillance sur le profil*;
QNorm,DQNorm :	*écart à un profil gaussien et incertitude*;

6.3 Limitations du logiciel MVM 1.0a

Nota Bene : La qualité des résultats dépend sensiblement de la cosmétique des images traitées dans la mesure où les coefficients en ondelettes impliqués mesurent les variations locales de contraste dans l'image. Aussi, avant de traiter une image avec le logiciel MVM, est-il fortement conseillée de vérifier l'absence de décrochements intempestifs d'intensité, notamment sur les bords.

La version actuelle du logiciel MVM souffre encore de plusieurs imperfections.

En particulier :

- les temps d'exécution ne sont pas optimisés ; pour une image de 4000×4000 pixels, la reconstruction par le canal des plans seuillés prend ainsi près de 3 heures de CPU sur un serveur Xéon bi-processeur cadencé à 2,2 GHz avec 8 Go de mémoire RAM.
- la reconstruction par arbre avec identification des structures a été développée dans le cadre d'un modèle de taches gaussiennes et ajusté afin d'offrir de bonnes performances dans les champs encombrés. Il en découle que la qualité de ce type de reconstruction est pour l'instant loin d'être optimale pour des profils beaucoup étendus qu'une gaussienne comme le montrent les tests fournis ci-après. En conséquence, les mesures associées à de telles structures ne seront au mieux qu'indicatives.

Chapitre 7

Applications 2D

7.1 Reconstructions de profils photométriques typiques

Les tests suivants ont tous été fait sur des images 2D et les résultats visualisés par une coupe 1D.

Les différents profils (d'intensité centrale 10000 et d'écart type 5) qui vont être étudiés avec les différents types de reconstruction sont présentés figure 7.1.

7.1.1 Stabilité de la reconstruction selon la procédure utilisée

Un premier test est fait pour mettre en évidence la stabilité de la reconstruction à partir d'un certain nombre d'itérations, et ce, quelque soit le profil étudié.

La procédure ***rec–image*** est utilisée pour ce test sur les différents profils sans bruit.

a) Cas d'une loi de Gauss
On constate d'après la figure 7.2 que la reconstruction pour un profil de gauss est stable au bout de 10 itérations et que l'écart entre le profil initial et la reconstruction n'est que de 0,25

b) Cas d'un loi de Moffat
Pour analyser un profil de Moffat, il a fallut soustraire une constante à l'image contenant ce profil afin de restreindre l'analyse sur 10 échelles ;

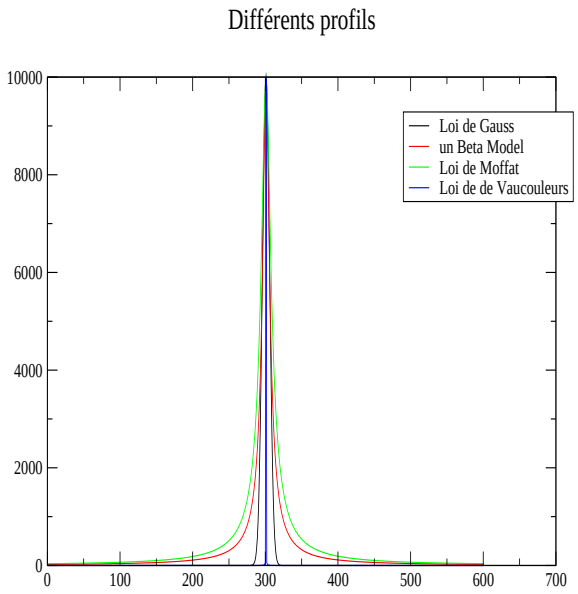


FIG. 7.1 – Superposition des différents profils d'intensité considérés pour modéliser les structures

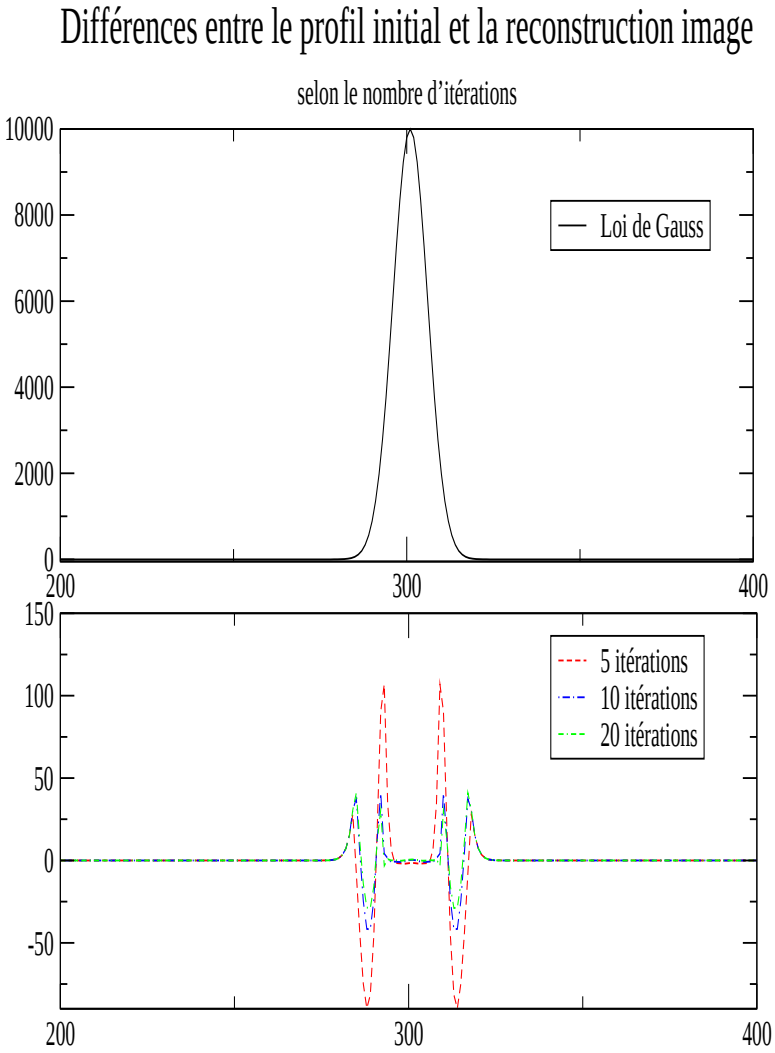


FIG. 7.2 – Reconstruction image d'un profil de Gauss

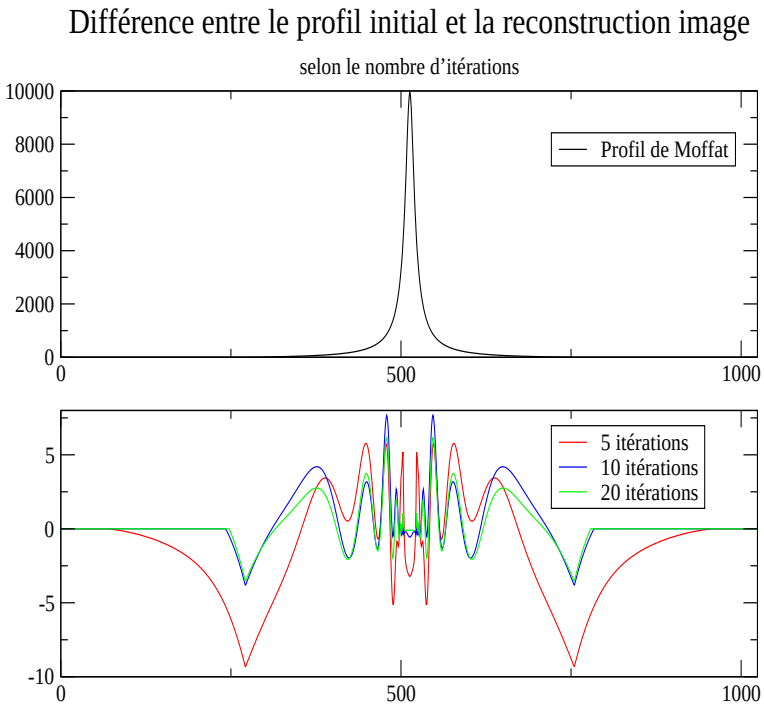


FIG. 7.3 – Reconstruction image d'un profil de Moffat

les discontinuités observées aux abcisses 270 et 760 s'expliquent par les discontinuités introduites au départ en soustrayant une constante.
On constate d'après la figure 7.3 que la reconstruction pour un profil de Moffat est à peu près stable au bout de 10 itérations.

c) Cas d'une loi de de Vaucouleurs
On constate d'après la figure 7.4 que la reconstruction pour un profil de de Vaucouleurs est à peu près stable au bout de 10 itérations et que l'écart entre le profil initial et la reconstruction n'est que de

d) Cas d'un Beta Model
Les mêmes discontinuités sont observées pour les mêmes raisons que pour le profil de Moffat. On constate d'après la figure 7.5 que la reconstruction pour Beta Model est à peu près stable au bout de 10 itérations.

Tous les tests présentés ci dessous ont été fait avec les différentes procédures de reconstruction avec 10 itérations et une erreur entre l'image reconstruite et l'image initiale de 0.001.

7.1.2 Qualité des reconstructions : Cas de la loi de Gauss

a) Profil de Gauss sans bruit additif

Le MVM est testé sur le même motif Gaussien, tout d'abord sans bruit, puis on introduiera un bruit gaussien additif pour controler la qualité de la détection en plus des performances de la restauration. Enfin, nous introduirons un bruit de poisson.
La figure 7.6 montre que les différentes reconstructions sont de bonne qualité sur un profil de Gauss.

b) Profil de Gauss avec bruit gaussien

Dans ce premier exemple, figure 7.7, on introduit un premier bruit gaussien, avec un SNR élevé (2) pour controler la qualité de la reconstruction.
On remarque que les reconstructions ***rec–image, rec–obj -A et rec–obj*** sont de très bonne qualité, et que les résidus entre l'image initiale bruitée et les reconstructions ne contiennent que du bruit. On peut constater sur la figure 7.8, que le résidu provient essentiellement du bruit.

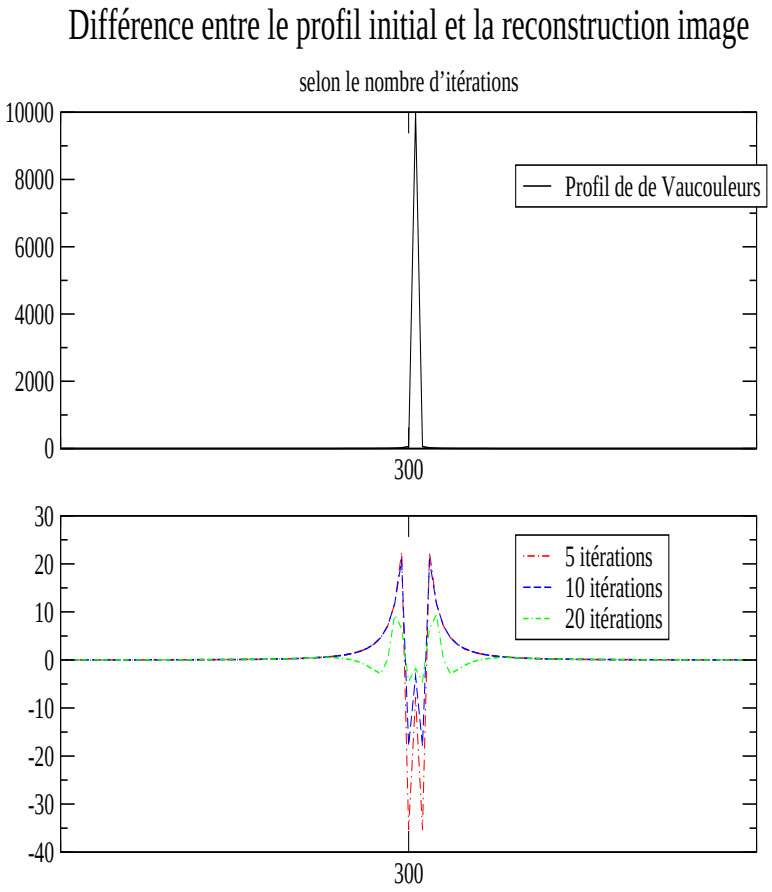


FIG. 7.4 – Reconstruction image d'un profil de de Vaucouleurs

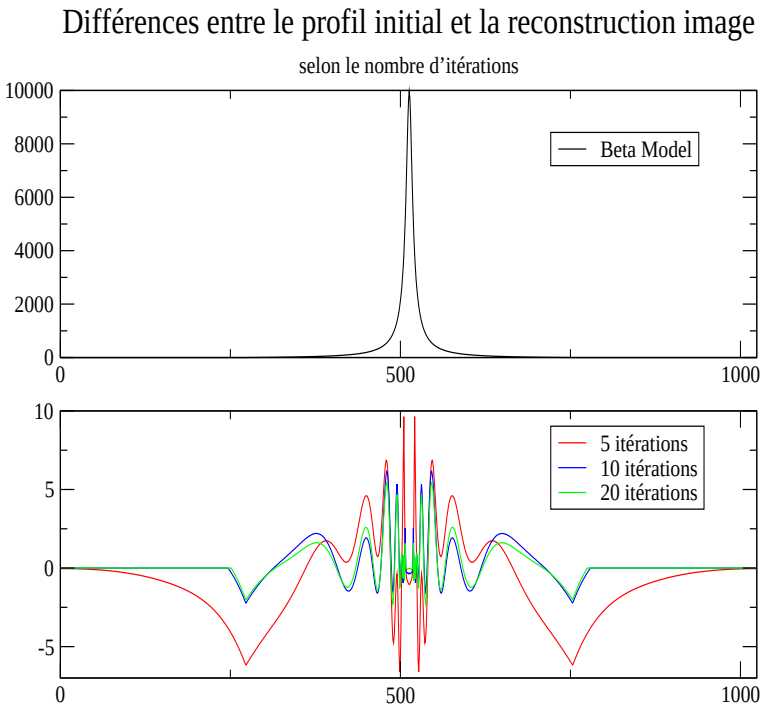


FIG. 7.5 – Reconstruction image d'un Beta Model

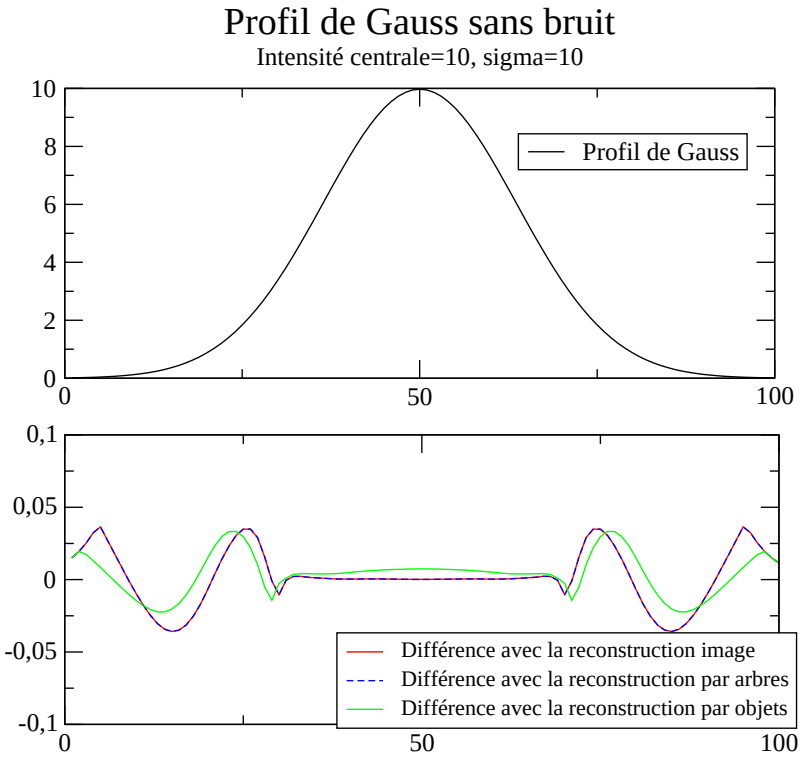


FIG. 7.6 – Différentes Reconstructions d'un profil de Gauss

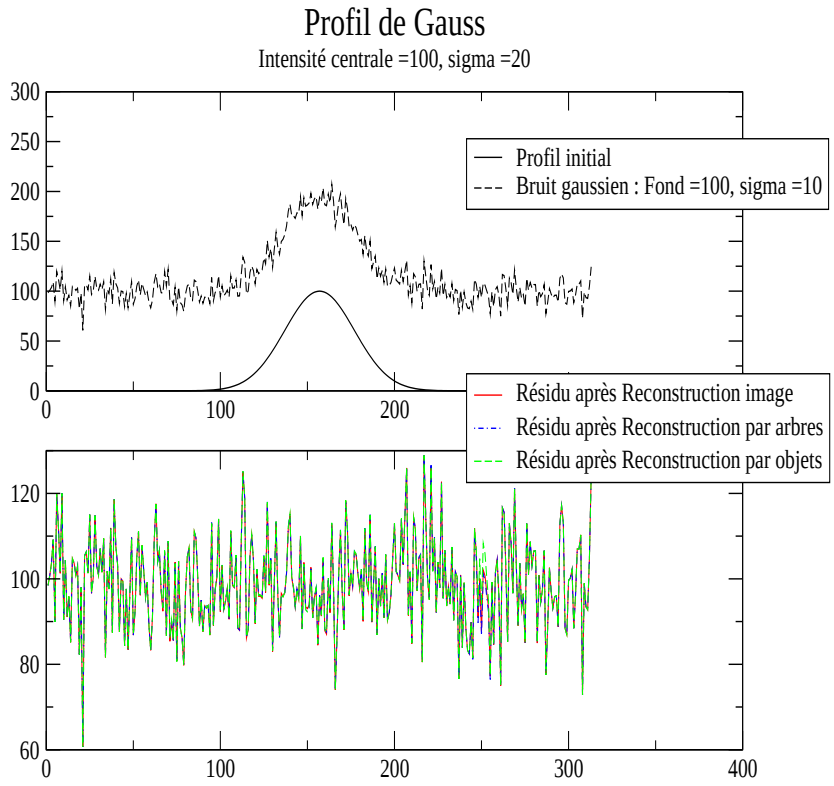


FIG. 7.7 – Différence entre les différentes reconstructions et le profil initial

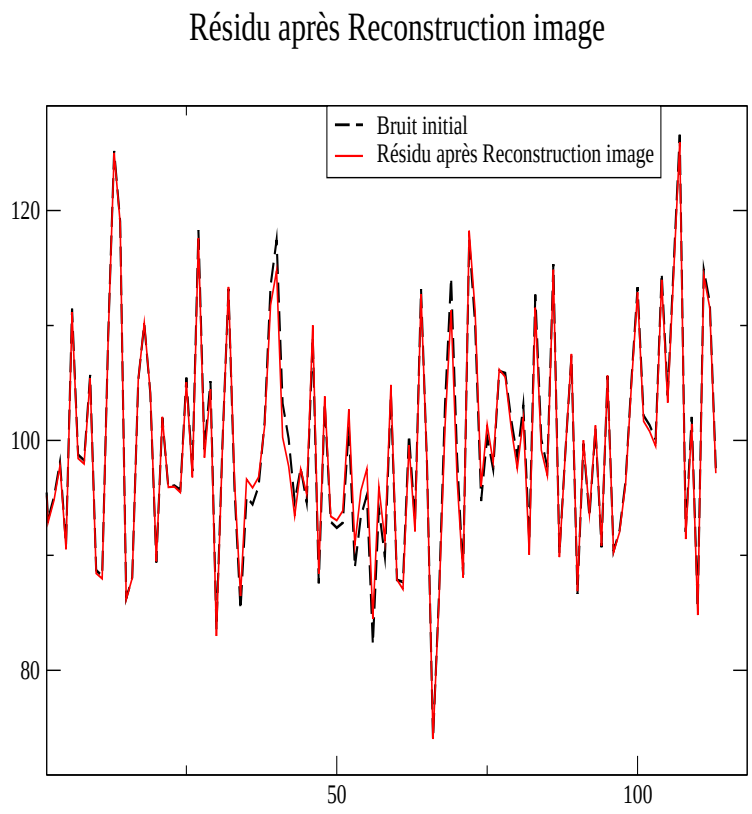


FIG. 7.8 – Différence entre le résidu de la Reconstruction image et une coupe de l'image de bruit

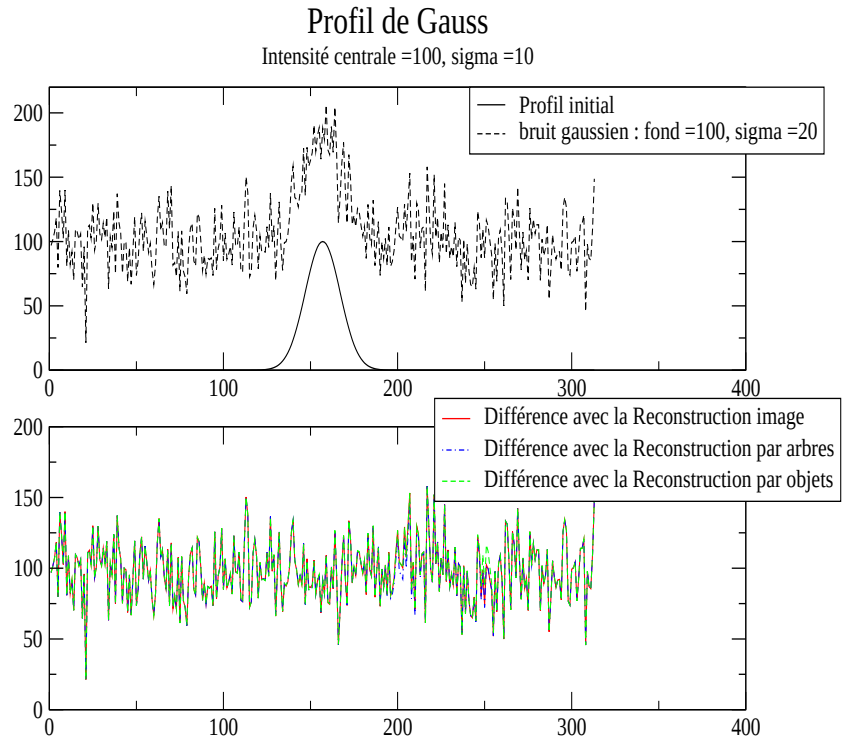


FIG. 7.9 – Différence entre les différentes reconstructions et le profil initial

Dans ce deuxième exemple, figure 7.9, on introduit un bruit gaussien, avec un SNR égale à 0.5.
On peut constater sur la figure 7.10, que le résidu provient essentiellement du bruit

Des tests ont montrés que lorsque l'objet à reconstruire a une intensité centrale très importante, le résidu observé montre des anneaux. Quelques tests ont montrés que ces anneaux disparaissent en prenant en compte dans notre analyse les coefficients négatifs. La figure 7.11 montre que la reconstruction avec les coefficients négatifs est de meilleure qualité.
La figure 7.12 montre les résidus dans les deux cas.
Par rapport à la dynamique, ces résidus restent très faibles (1/1000)

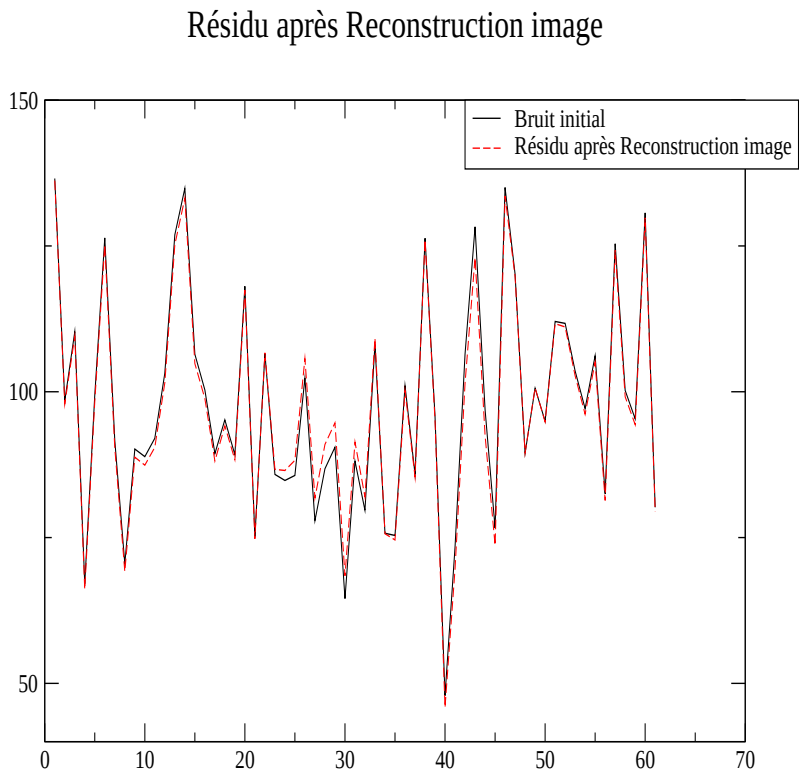


FIG. 7.10 – Différence entre le résidu de la Reconstruction image et une coupe de l'image de brui

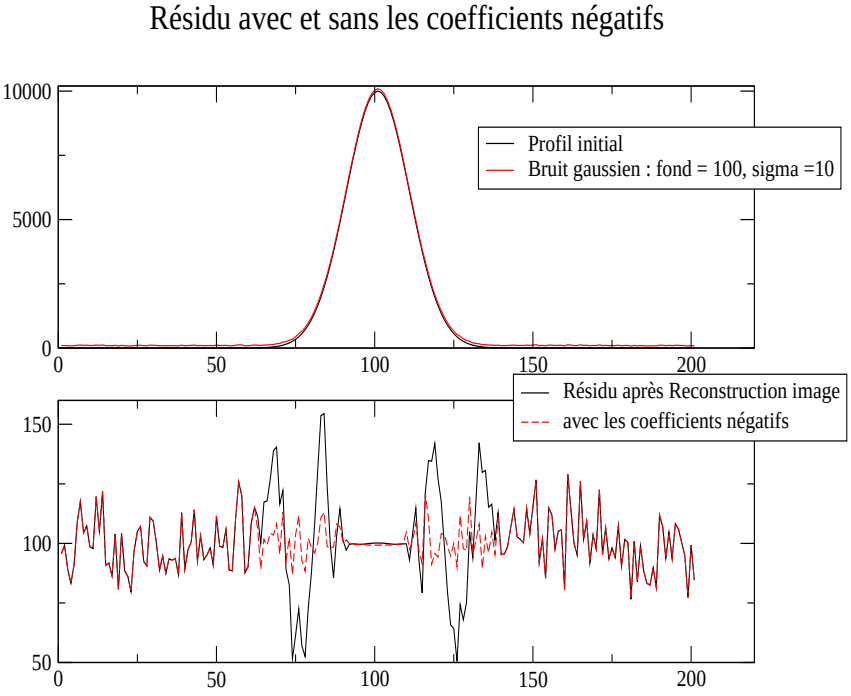


FIG. 7.11 – Reconstructions avec et sans les coefficients négatifs

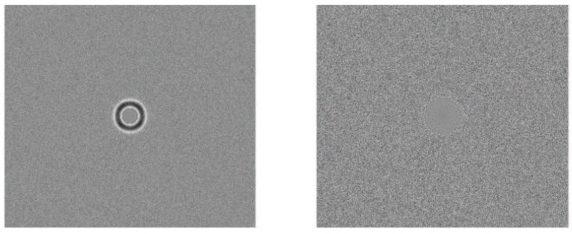


FIG. 7.12 – A gauche Résidu sans les coefficients négatifs, à droite avec les coefficients négatifs

et sont comparables à la précision demandée lors de la reconstruction ; statistiquement, l'intégrale de flux est juste.
Pour vérifier statistiquement qu'une reconstruction est meilleure que l'autre (avec ou sans coefficients négatifs), nous allons comparer les variances avec le test de snedecor.
Si l'on calcul l'écart type (pour la ligne 256 correspondant à la coupe effectuée sur les deux résidus), on constate que celui du résidu calculé avec les coefficients négatifs est inférieur au résidu d'origine.
Le test de Fisher-Snedecor va nous permettre de comparer les variances pour un seuil de refus à un risque donné (risque qu'un effet soit considéré comme significatif alors qu'il ne l'est pas).

valeur des carts type :

pour le résidu calculé à partir des coefficients négatifs : $\sigma_B = 9.8155$
pour le résidu d'origine : $\sigma_A = 13.1165$

Pour valider statistiquement que $\sigma_B < \sigma_A$, on va utiliser le ***test de snedecor*** :

Conditions de validités : les populations sont normales

on a donc deux populations A et B et 255 degrés de libertés α_1 et α_2
on calcul les variances :
$s_A^2 = 172.043$
$s_B^2 = 96.344$

hypothèse H0 : $s_B^2 <= s_A^2$

F $= s_B^2/s_A^2 = 96.344/172.043 = 0.56 < 1$ (valeur de la table de Snedecor pour un risque de 1%)

On admet donc que $\sigma_B <= \sigma_A$ avec un risque de 1%.

On veut démontrer que $\sigma_B < \sigma_A$.

Hypothèse : $s_B^2 = s_A^2$

F= 172.043/96.344 = 1.7857
pour un risque de première espèce $\alpha = 0.01$
$\alpha_1 = \alpha_2 = 0.005$

$F(1-\alpha_2) = F_0.99 = 1$
$F\alpha_1 = F_0.005 = 1/F_0.99 = 1$

donc F ne vérifie pas $F\alpha_1 < \mathrm{F} < F(1-\alpha_2)$
par conséquent on rejète l'hypothèse $s_B^2 = s_A^2$ avec un risque de première espèce de 1%.

Conclusion : on a bien démontré statistiquement que $\sigma_B < \sigma_A$, donc que **le résidu provenant de la TO avec les coefficients négatifs est moindre**.

c) Profil de Gauss avec bruit de poisson

Dans cet exemple, nous introduisons un bruit de poisson pour voir la qualité de la détection et de la reconstruction avec les trois routines ***rec–image***, ***rec–obj -A*** et ***rec–obj***.

Les différentes reconstructions pour une gaussienne dans le cas d'un bruit de poisson sont illustrées figure 7.13.
Deux exemples sont illustrés figures 7.14 et 7.15 et compare les différentes reconstructions images avec et sans les coefficients négatifs dans le cas d'une gaussienne d'intensité centrale 100 et d'une gaussienne d'intensité centrale 10000.
La figure 7.16 montre les résidus dans les deux cas.

7.1.3 Qualité des reconstructions : Cas des autres lois

a) Loi de Moffat sans bruit additif
La reconstruction d'un profil de Moffat est de meilleure qualité si l'on prend en compte dans l'analyse les coefficients négatifs. La figure 7.17 montre les résidus entre le profil initial et la reconstruction image avec et sans les coefficients négatifs.
Les différentes reconstructions d'un profil de Moffat sont illustrées figure 7.18 et montre que la reconstruction objet est encore de mauvaise qualité.

b) Loi de de Vaucouleurs sans bruit additif
La reconstruction d'un profil de de Vaucouleurs est de meilleure qualité

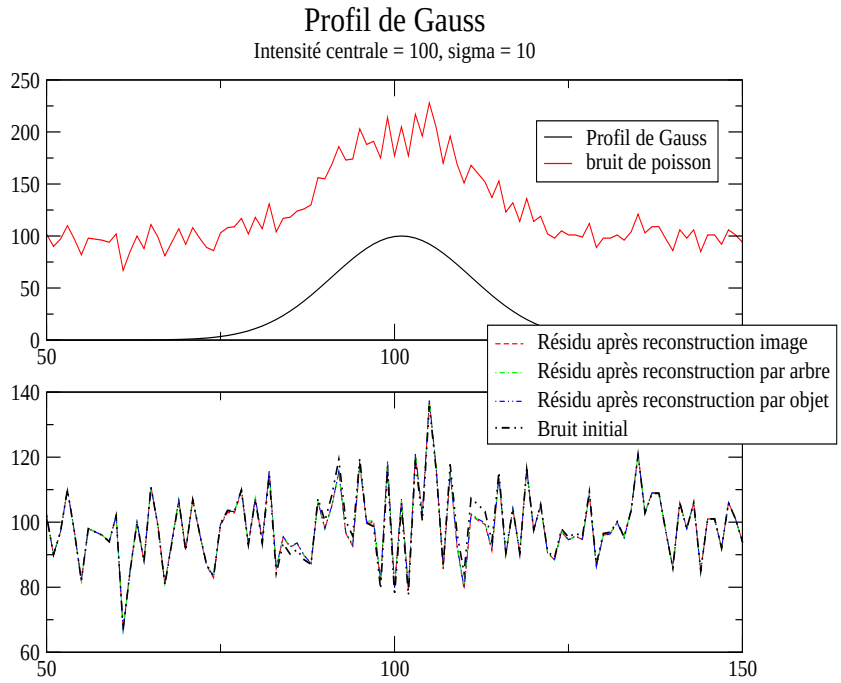


FIG. 7.13 – Bruit de Poisson et différentes Reconstructions

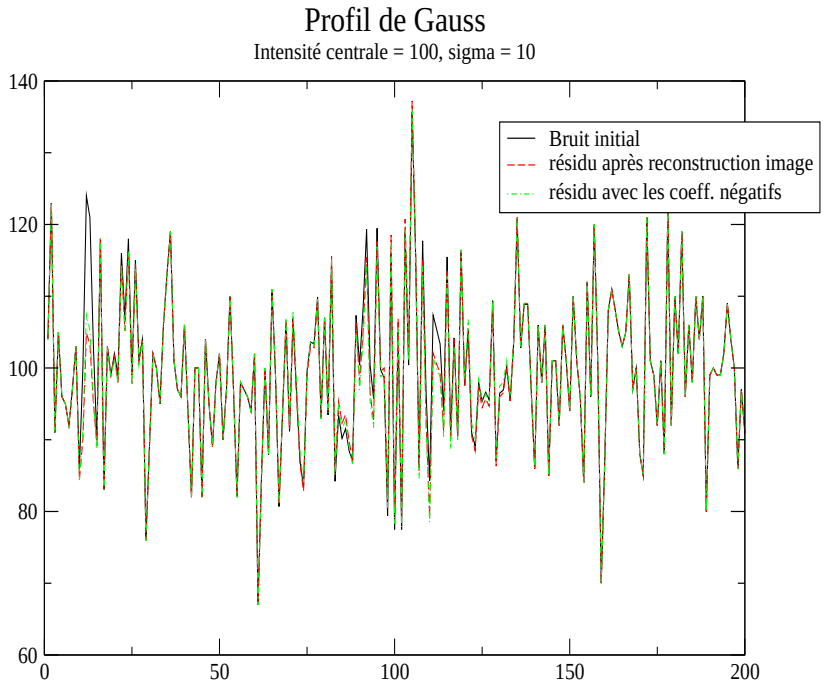


FIG. 7.14 – Résidus avec ou sans les coefficients négatifs

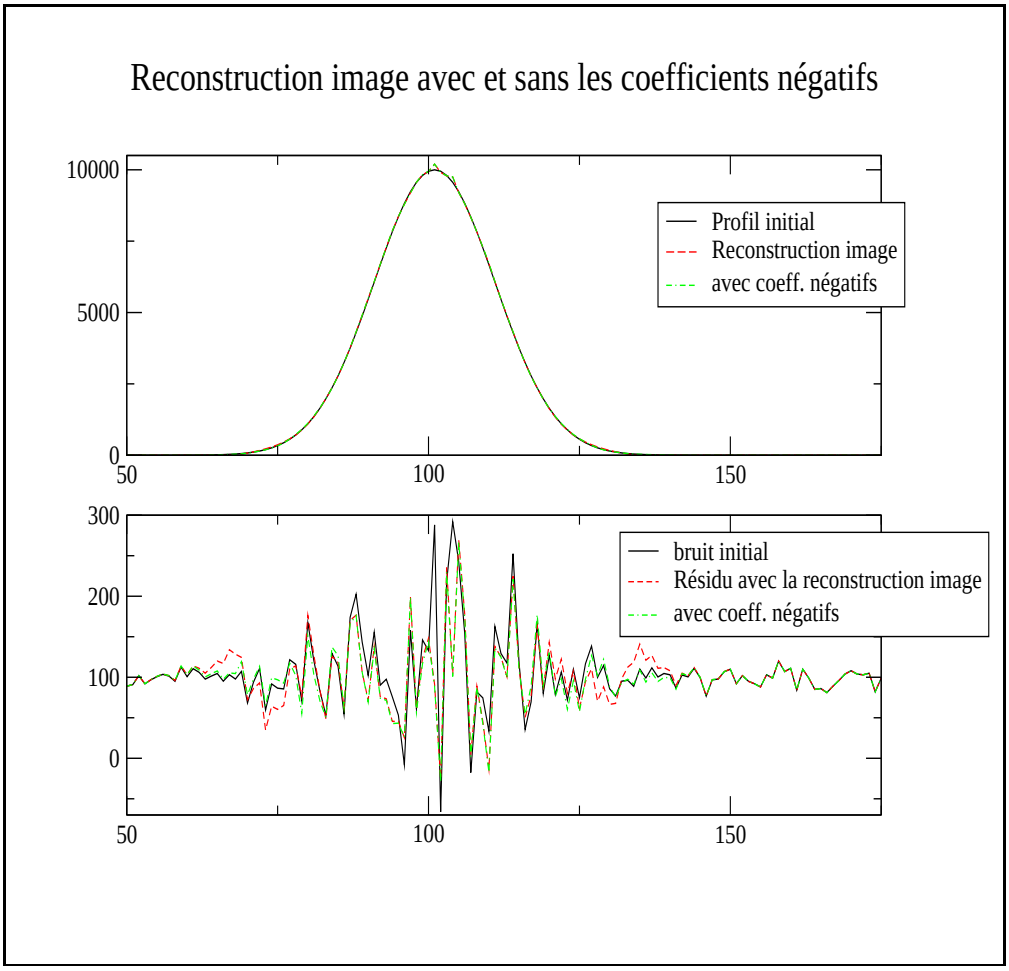


FIG. 7.15 – Résidus avec ou sans les coefficients négatifs

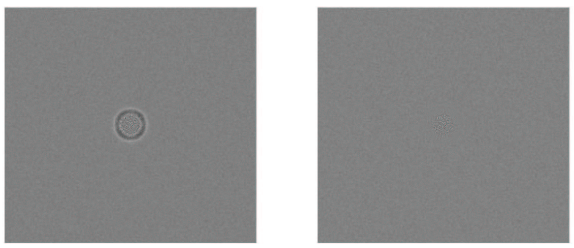


FIG. 7.16 – A gauche Résidu sans les coefficients négatifs, à droite avec les coefficients négatifs

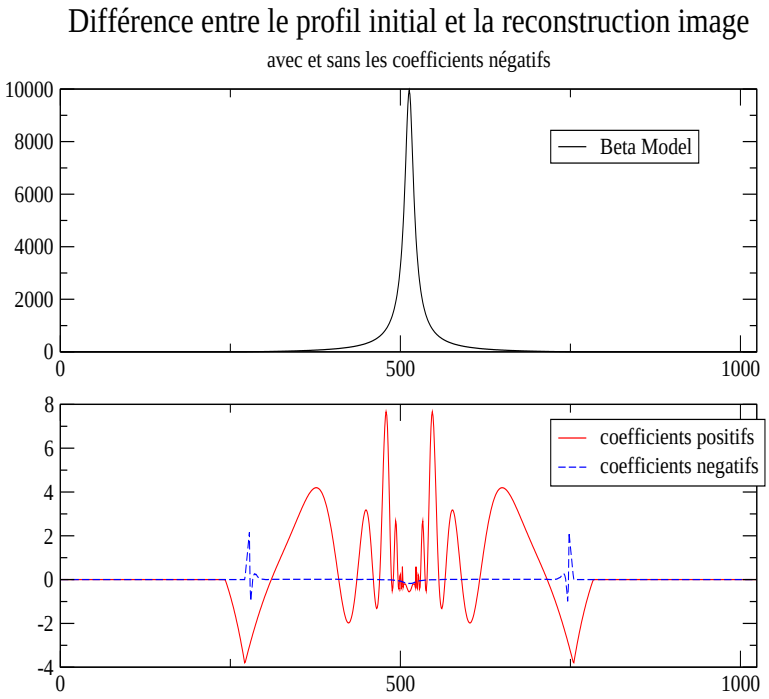


FIG. 7.17 – Profil de Moffat

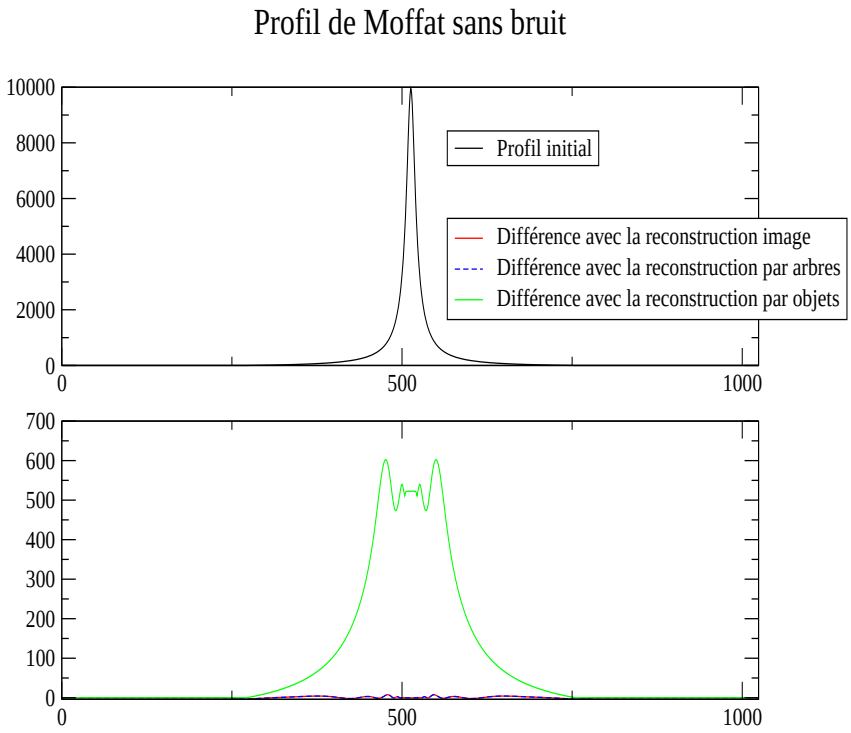


FIG. 7.18 – Différentes reconstructions d'un profil de moffat

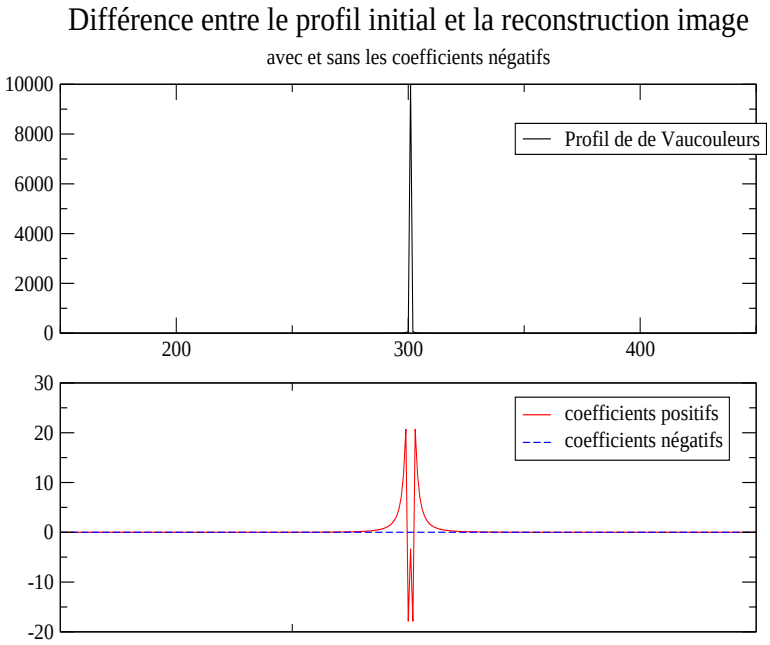


FIG. 7.19 – Profil de de Vaucouleurs

si l'on prend en compte dans l'analyse les coefficients négatifs. La figure 7.19 montre les résidus entre le profil initial et la reconstruction image avec et sans les coefficients négatifs.
Le MVM est testé sur un profil de de Vaucouleurs sans bruit pour controler la qualité de la détection en plus des performances de la restauration.

c) un β Model sans bruit additif
La reconstruction d'un Beta Model est de meilleure qualité si l'on prend en compte dans l'analyse les coefficients négatifs. La figure 7.21 montre les résidus entre le profil initial et la reconstruction image avec et sans les coefficients négatifs.
Les différentes reconstructions d'un Beta Model sont illustrées figure 7.22 et montre que la reconstruction objet est encore de mauvaise qualité.

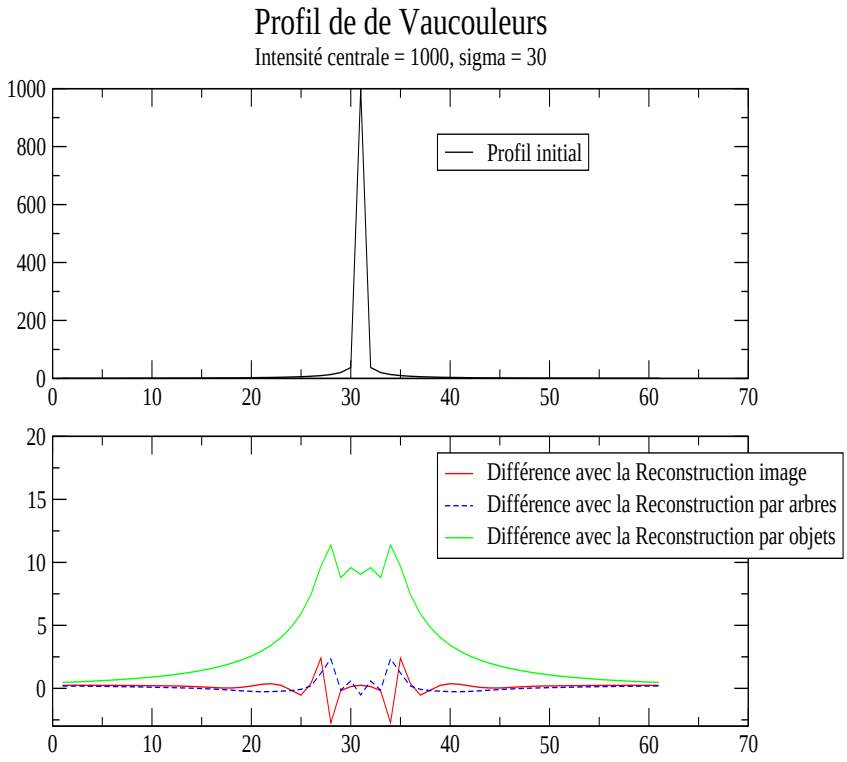


FIG. 7.20 – Différentes Reconstructions d'un profil de de Vaucouleurs

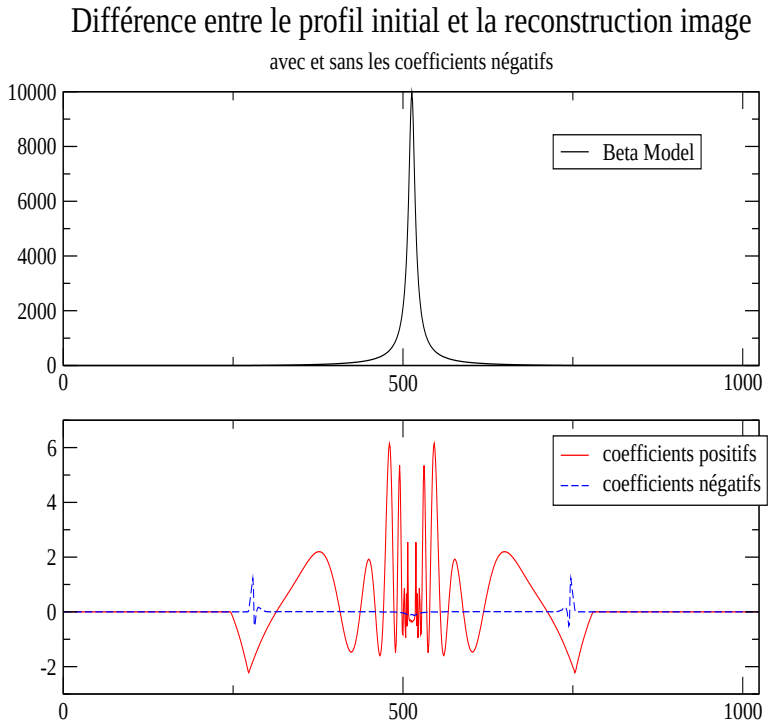


FIG. 7.21 – Beta Model

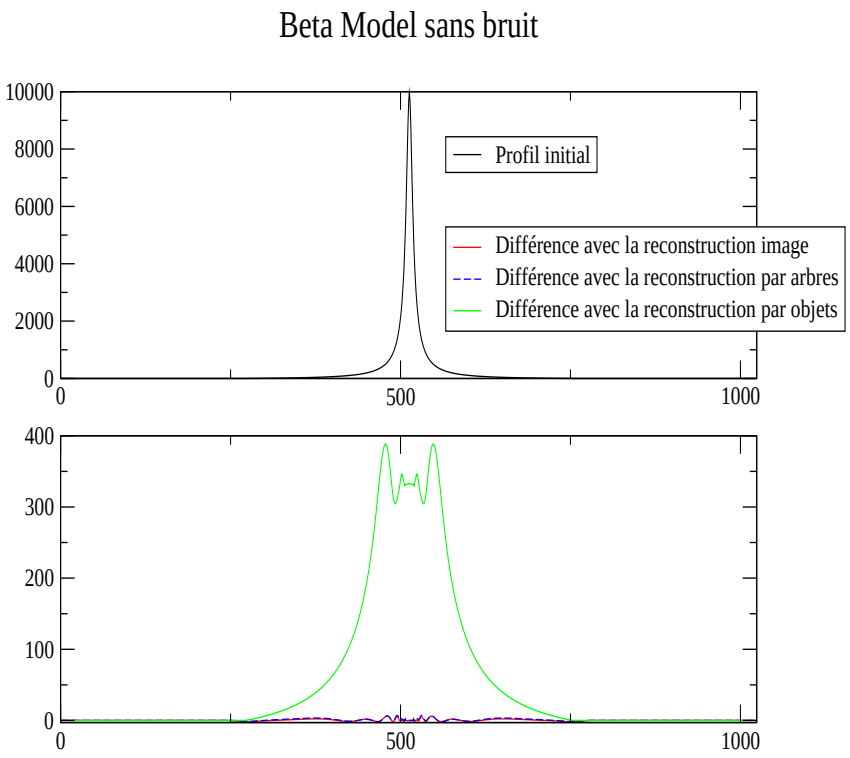


FIG. 7.22 – Différentes reconstructions d'un Beta Model

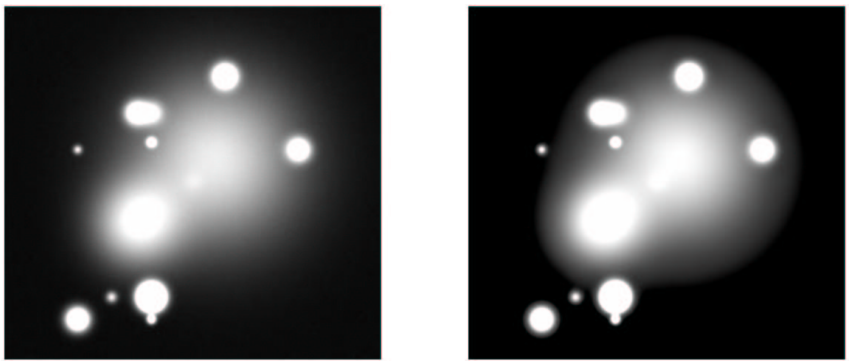


FIG. 7.23 – A gauche Profils initiaux, à droite Reconstruction image

7.1.4 Reconstruction de plusieurs profils

La reconstruction de plusieurs profils est illustrée par la reconstruction image.

a) Premier cas

L'image figure 7.23 représente plusieurs gaussiennes d'intensité et d'écart type différents avec un bruit gaussien. La reconstruction image comme le montre la figure 7.23 est de bonne qualité.

b) Deuxième cas

Lorsque les profils à reconstruire s'écartent des profils précédents ou d'intensité centrale très différentes, on observe sur l'image reconstruite que les profils d'intensité importante sont mieux reconstruits.
L'exemple de la figure 7.24 illustre le cas où l'algorithme de reconstruction montre ses limites dans la version actuelle.
Ces résultats viennent du fait que les paramètres de convergence de l'algorithme de reconstruction sont calculés à chaque itération globalement ; ces paramètres vont donc être mal définis pour l'objet de type gaussien au profit de l'objet plus intense.

Conclusions :
Un objet carré sera moins bien reconstruit qu'un objet dont la forme est proche de l'ondelette analysante. Un objet faible sera pénalisé pour la reconstruction par rapport à un objet de forte intensité. La reconstruction de ce type d'objet engendre un anneau négatif qui se trouve visible par le décrochement qu'il provoque.

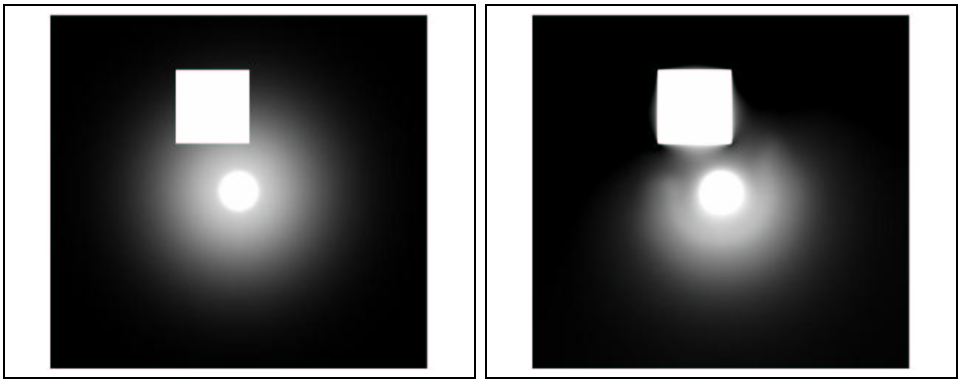


FIG. 7.24 – A gauche Profils initiaux, à droite Reconstruction image

Ceci illustre l'intérêt d'une reconstruction par objet qui va calculer ces paramètres par objet et donc optimiser les résultats pour chaque objet reconstruit.

Conclusions et Perspectives

Annexes

7.2 Algorithme à trous

7.3 Outils de visualisations du graphe des structures identifiées

Outils de visualisations brutes :

Les outils de visualisation des images au format FITS et des plans en ondelettes au format FITS sont les outils standard tels que : saoimage, ds9...

Pour la visualisation du graphe d'objets *name.gro.fits*, sortie du programme ***obj–wave***, les outils sont les suivants :

7.3.1 *voir–foret*

Le programme ***voir–foret*** permet d'obtenir un schéma sommaire de l'arborescence des objets détectés dans l'image sous forme échelle, numéro d'objet.

Commande :	*voir–foret* name *[-A]*	*-A]* : option permettant de visualiser l'arborescence du graphe d'arbres (c'est à dire avant identification des objets)
Entrée :	*name.gro.fits* ou *name.gra.fits*	graphe d'objets ou graphe d'arbres
Sortie :	*écran*	arborescense des objets et sous objets identifiés ou non identifiés à l'écran

Exemple : *voir–foret* L384 (si et seulement si L384.gro.fits existe dans le même répertoire) *voir–foret* L384 *-A* (si L384.gra.fits existe dans le même répertoire)

On obtient à l'écran l'arborescence des objets identifiés.(sortie ascii)

7.3.2 complexe

Commande :	*complexe name.gro.fits*	
Entrée :	*name.gro.fits*	
Sortie :	Ecran	pour chaque échelle, identification de la complexité d'un objet (écran) (un objet est complexe si il a au moins un sous objet)

Exemple : *complexe* L384.gro.fits

Cet outil va permettre de choisir entre les stratégies d'analyse selon la nature des objets.

7.3.3 *carte–obj*

Commande :	*carte–obj name [-S] [-num] [-L]*
Entrée :	*name.gro.fits*
Sortie :	carte des objets et sous–objets identifiés (au format ps)

Description des options :

-S :	ne fait apparaître que les sous objets sans numéro ;
-num :	avec numéro d'objets
-L :	liste d'objets sortie de *make–liste*

Exemple : *carte–obj* L384 *-num* (ne fait apparaître que les objets avec leurs numéros ; voir figure 7.25) *carte–obj* L384 *-num -S* (ne fait apparaître que les sous objets avec leurs numéros) *carte–obj* L384 *-num -L* (ne fait apparaître que les objets de la liste L)

Cet outil ne peut être utilisé que si le nombre d'objets à visualiser n'est pas trop important.

7.3.4 *dessine–arbre*

Une fois les objets complexes repérés, on peut obtenir un graphe de leur arborescense à l'aide du programme dessine–arbre.

Commande :	*dessine–arbre name.gro.fits*
Entrée :	*name.gro.fits*
Sortie :	fichier postcript contenant le dessin de l'objet sélectionné

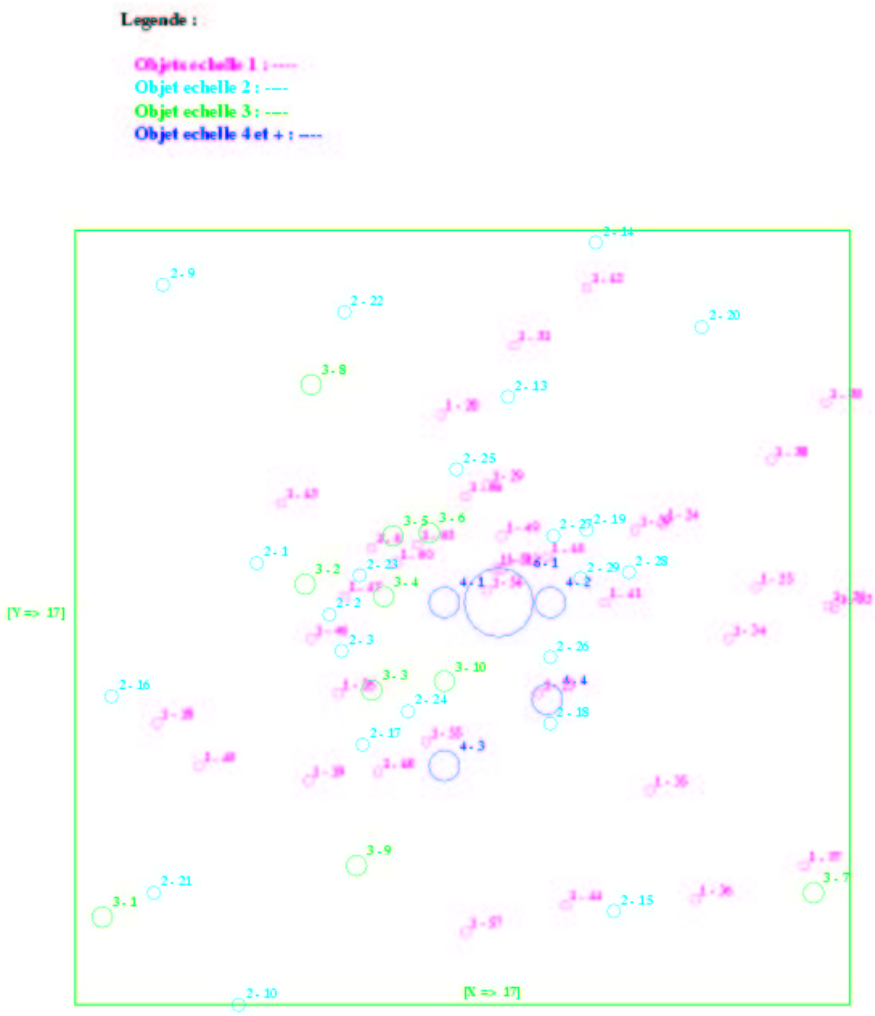


FIG. 7.25 – Carte des objets identifiés dans L384.fits

7.3.5 *make–liste*

Cet outil permet de gérer la série d'objets à reconstruire, donc de créer une liste d'objets et sous–objets que l'on donnera en paramètre à *rec–obj* par la suite. Il s'agit d'un programme interactif permettant de choisir les objets selon plusieurs options.

Commande :	*make–liste name.gro.fits*
Entrée :	*name.gro.fits*
Sortie :	liste *name.lst* (selon l'option suivante choisie)

1- Visualisation de l'arbre des objets
2- Visualisation des attributs de reconstruction actuels
3- Introduction de la nature des objets
4- Selection de tous les objets
5- Selection des objets par complexite
3- Selection a partir d'une liste (déjà créée précédemment)6
7- Selection par objet
8- Selection par echelle
9- Selection par nature
10- Sauver
11- Quitter

Commande :	*make–liste name.gro.fits*
Entrée :	*name.gro.fits*
Sortie :	liste *name.lst*

Exemple : *make–liste* L384.gro.fits objectif : ne reconstruire qu'un seul objet numéro i à l'échelle j

Etapes :
4- Sélection de tous les objets
0- Pas de reconstruction et exclu des statistiques
7- Sélection par objet Pour quelle échelle souhaitez-vous modifier les attributs ? Echelle : j Numero : i Portée des modifications :
1- Attribut de l'objet
2- Attributs des sous objets de l'objet ensuite on choisi l'option 2 : reconstruction et intégration statistique ensuite 10 Sauver : L384.lst

Cette liste sera ensuite donnée comme paramètre d'entrée à rec–obj *(rec–obj L384 -L L384.lst)* pour ne reconstruire que les objets de celle-ci.

7.4 Evaluation morphologique et photométrique : Catalogue d'objets

Bibliographie

[1] J.Morlet, G.Arens, E.Fourgeau, and D.Giard. Wave propagation and sampling theory - part i, ii. *Geophysics*, 47(2) :203-236, Febuary 1982.

[2] M. Holdschneider, R. Kronland-Martinet, J. Morlet, and P. Tchamitchian. Wavelets : Time-Frequency Methods and Phase-Space, chapter A Real-Time Algorithm for Signal Analysis with the Help of the Wavelet Transform, pages 286-297. Springer Berlin, 1989.