Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Institut National de formation en Informatique (I.N.I) Oued-smar Alger

Mémoire de fin d'études

En vue de l'obtention du diplôme d'Ingénieur d'État en Informatique

Option : Systèmes Informatiques

Multipass : Guide d'Installation, d'Utilisation et d'Extension

Réalisé par : M^r Anis BENYELLOUL M^r Said-Eddine BENSALEM Proposé par : $M^r M. KOUDIL$ $M^r B. SOUICI$

Promotion : 2006/2007

Table des matières

1	Intr	oduction	2
2	Pré	sentation de GIMP	3
	2.1	Introduction	3
	2.2	Concepts fondamentaux de GIMP	4
	2.3	La boîte à outils principale	5
	2.4	Fenêtre d'image	6
	2.5	Dialogues, fenêtres et onglets	9
3	Util	isation de Multipass	12
	3.1	Mode Image Unique	12
	3.2	Mode «Batch»	16
4	\mathbf{Ext}	ension de Multipass	19
	4.1	Le fichier d'entête	19
	4.2	Le fichier source	21
5	Con	npilation et Installation de Multipass	24
	5.1	Compilation	24
	5.2	Installation	25

Chapitre 1

Introduction

Ce manuel constitue un guide d'utilisation, d'extension, et d'installation de Multipass. Multipass est une application implémentant des algorithmes biomimétique de segmentation d'images. Le programme se présente sous la forme d'un greffon (ou additif, ou «plugin») pour le logiciel de traitement d'image GIMP.

Ce document contient les chapitre suivants :

- Chapitre 2 : Présentation de GIMP : Ce chapitre contient une introduction à GIMP lui-même. Une bonne maîtrise de GIMP est indispensable pour pouvoir utiliser Multipass efficacement. Pour une présentation plus complète voir www.gimp.org.
- Chapitre 3 : Utilisation de Multipass : Ce chapitre est un guide d'utilisation de Multipass lui même. Il montre comment se servir de son interface et comment interpréter les résultats.
- **Chapitre 4 : Extension de Multipass :** Ce chapitre est consacre à «l'extension de Multipass», c-à-d à la procédure à suivre pour étendre ses fonctionnalités et ajouter ne nouveaux algorithmes de segmentation.
- Chapitre 5 : Compilation et Installation de Multipass : Le dernier chapitre montre comment recompiler et installer Multipass a partir de son code source.

Chapitre 2

Présentation de GIMP

2.1 Introduction

GIMP est un outil portable de manipulation d'images et de photos. Le mot GIMP est un acronyme de GNU Image Manipulation Program, en français Programme GNU de Manipulation d'Images. GIMP est utilisable pour une grande variété de taches de manipulations d'images comme la retouche photo, la composition ou la création d'images.

Une des forces de GIMP est sa libre disponibilité depuis de nombreuses sources pour de nombreux systèmes d'exploitations. La plupart des distributions GNU/Linux incluent GIMP en standard dans leurs applications. GIMP est aussi disponible pour d'autres systèmes d'opérations comme Microsoft Windows ou Mac OS X d'Apple (Darwin). Plus important, GIMP n'est pas un freeware, c'est un logiciel Open Source couvert par la licence GPL. La GPL donne à l'utilisateur la liberté de lire et de modifier le code source qui compose le programme.

Voici un condensé des fonctionnalités de GIMP :

- Suite complète d'outils de dessin comprenant un crayon, un pinceau, un aérographe, etc.
- Gestion partitionnée de la mémoire de telle façon que la taille de l'image ne soit limitée que par l'espace disque disponible.
- Support de la transparence;
- Une suite de procédures pour appeler les fonctions internes de GIMP à partir de programmes extérieurs;
- Historique d'annulation illimité;
- Outils de transformations tels que rotation, étirer/rétrécir, cisaillement et miroir;
- Nombreux format de fichiers supportés GIF, JPEG, PNG, XPM, TIFF, TGA, MPEG, PS, PDF, PCX, BMP, etc;
- Outils de sélection tels que rectangle, ellipse, main levée, par couleur, contiguë, courbes de bézier et ciseaux intelligents



FIG. 2.1 – Arrangement des fenêtres dans GIMP

2.2 Concepts fondamentaux de GIMP

La copie d'écran de la figure 2.1 montre l'arrangement le plus simple des fenêtres les plus utilisées de GIMP (Cette configuration est celle par défaut légèrement simplifiée). Ces trois fenêtres sont :

- 1. La boîte à outils principale : C'est la fenêtre principale de GIMP. Elle contient le menu principal, un ensemble d'icônes servant à sélectionner les outils, etc.
- 2. Les options des outils : Sous la boîte à outils se trouve accolé le dialogue des Options des outils, il affiche les options de l'outil sélectionné.
- 3. Une fenêtre d'image : Chaque image ouverte par le GIMP est affichée dans une fenêtre séparée. Plusieurs images peuvent être ouvertes en même temps : la limite n'est imposée que par vos ressources systèmes.
- 4. Le dialogues des calques : Cette boite de dialogue montre la structure des calques de l'image active et permet de les manipuler de diverses manières.
- 5. Brosses, Motifs, Dégradés : Le dialogue à onglets rattaché sous le dialogue des calques contient les différents dialogues pour gérer les brosses, motifs et dégradés.

C'est la configuration minimale. Il y a plus d'une douzaine d'autres dialogues dans GIMP pour divers usages, mais les utilisateurs les ouvrent au besoin et les referment après utilisation. Les utilisateurs gardent généralement la boîte à outils (avec les options des outils) et le dialogue des calques ouverts en permanence. La boîte à outil est essentielle pour de nombreuses opérations de GIMP; d'ailleurs, si vous la fermez, GIMP s'arrêtera (vous devrez confirmer que c'est bien là ce que vous voulez faire). Le dialogue Options des outils est maintenant un dialogue à part, montré ici accolé sous la boîte à outils. C'est ainsi que la plupart des utilisateurs configurent ces dialogues : il est très difficile d'utiliser efficacement les outils sans voir l'état des options qui leur sont associées. Le dialogue des calques intervient dès que vous travaillez avec une image qui contient plusieurs calques. Enfin, il est nécessaire d'avoir des images affichées pour travailler dessus.

Si vos dispositions des fenêtres GIMP sont modifiées, l'arrangement montré dans la copie d'écran de la figure 2.1 peut etre très facilement rétabli. Dans le menu Fichier de la boîte à outil sélectionnez Fichier->Dialogues->Nouvelle fenêtre attachable->Calques, canaux et chemins pour afficher un dialogue des calques comme ci-dessus. Dans le même menu, sélectionnez Fichier->Dialogues->Options des outils pour afficher un nouveau dialogue Options des outils que vous pourrez accoler à la fenêtre principale (la section Accolement explique comment accoler des dialogues). Il n'est pas nécessaire de créer une nouvelle boîte à outils principale puisque vous ne pouvez pas fermer celle que vous avez sans quitter GIMP.

Contrairement à d'autres programmes, GIMP ne vous permet pas de tout regrouper (Dialogues et fenêtres image) dans une seule fenêtre principale. Les développeurs de GIMP ont toujours pensé que c'était une mauvaise façon de travailler, parce que cela force le programme à exécuter un certain nombre de fonctions qui sont bien mieux réalisées par un gestionnaire de fenêtres. Non seulement cela ferait perdre beaucoup de temps aux programmeurs mais ce serait presque impossible à faire fonctionner correctement sous tous les systèmes d'exploitations pour lesquelles GIMP est opérationnel.

Les versions précédentes de GIMP (jusqu'à 1.2.5) étaient très dispendieuses de dialogues : les utilisateurs avaient souvent une demi-douzaine de dialogues ouverts en même temps, éparpillés sur l'écran et il était difficile de les garder en vue. GIMP 2.0 est bien moins fouillis puisqu'il permet d'accoler les dialogues de manière flexible (le dialogue des calques de la copie d'écran contient en fait quatre dialogues, représentés par des onglets : calques, canaux, chemins et historiques d'annulation).

Les sections suivantes décriront les composants de chaque fenêtre de la copie d'écran de la figure 2.1 pour vous expliquer ce qu'ils sont et comment s'en servir. Quand vous les aurez lues, ainsi que la section décrivant la structure de base des images du GIMP, vous en saurez assez pour utiliser le GIMP pour de nombreuses manipulations simples d'image.

2.3 La boîte à outils principale

La boîte à outils est le cœur de GIMP. C'est la seule partie de l'application que vous ne pouvez ni dupliquer ni fermer. Voici un rapide aperçu de ce que vous y trouverez (voir figure 2.2) :

- Le menu de la boîte à outil : Ce menu est particulier : il contient des commandes que l'on ne retrouve pas dans les menus des fenêtres image (et d'autres qui y sont). Il s'agit des commandes qui permettent de paramétrer les préférences, de créer certain type de dialogues, etc.
- 2. Les icônes outils : Ces icônes sont des boutons qui activent les outils permettant d'effectuer de nombreuses manipulations telles que : sélectionner des parties d'image, les peindre, les transformer, etc.



FIG. 2.2 – La boîte à outils principale de GIMP

- 3. Couleurs de premier-plan et d'arrière-plan : Les aires de couleurs vous montrent les couleurs de premier plan et d'arrière plan actuelles, elles interviennent dans de nombreuses opérations. Cliquer sur l'une d'entre elles fera apparaître un dialogue de choix des couleurs qui vous permettra de la changer. Cliquer sur la double-flèche interchange les deux couleurs, et cliquer sur le petit symbole dans le coin en bas à gauche réinitialise le noir et le blanc.
- 4. Brosse, Motif et Dégradé : Ces symboles vous montrent la sélection actuelle de GIMP pour : la brosse, utilisée par tous les outils qui permettent de dessiner sur l'image ("dessiner" est pris au sens large et inclut les opérations telles que le gommage et le barbouillage); le motif, utilisé pour remplir des aires sélectionnées d'une image; et le dégradé, qui intervient dès qu'une opération nécessite une transition de couleurs. Un clic sur l'un de ces symboles affichera le dialogue permettant d'en changer.
- 5. L'image active : Nouvelle fonctionnalité apparue dans GIMP 2.2. Dans GIMP, vous pouvez travailler avec plusieurs images en même temps, mais à chaque instant, l'une d'entre elle est "l'image active". Vous trouverez ici une représentation miniature de cette image. Cliquer sur elle ouvrira un dialogue listant toutes les images actuellement ouvertes, vous permettant de changer l'image active si vous le désirez (on peut aussi rendre une image active en cliquant sur la fenêtre de l'image).

2.4 Fenêtre d'image

Dans GIMP, chaque image ouverte est affichée dans sa propre fenêtre (dans certain cas la même image peut être affichée dans plusieurs fenêtres mais c'est inhabituel). Nous commencerons par une brève description des composants présents dans une fenêtre image ordinaire (voir figure 2.3) :



FIG. 2.3 – Fenêtre d'image de GIMP

- 1. Barre de titre : En haut de la fenêtre image se trouve probablement la barre de titre dans laquelle s'affiche le nom de l'image et quelques informations de base. La barre de titre est en fait générée par le gestionnaire de fenêtre du système et non par GIMP, aussi son apparence peut changer selon le système d'exploitation, le gestionnaire de fenêtre et/ou le thème. Dans le Dialogue des préférences vous pouvez personnaliser les informations qui y apparaîssent.
- 2. Barre des Menus : Juste en dessous de la barre de titre se trouve le «Menu d'image» (à moins qu'il n'ait été désactivé). Ce menu vous permet d'accéder à presque toutes les opérations réalisables sur une image (Il y a des actions "globales" qui ne sont accessibles que par le menu de la boîte à outil). Vous pouvez aussi dérouler le menu image en cliquant avec le bouton droit sur l'image, ou bien en cliquant sur la flèche dans le coin supérieur gauche, si l'une de ces méthodes vous convient mieux. De plus, la plupart des opérations du menu peuvent aussi être activées depuis le clavier, en utilisant Alt et une touche "accélératrice" soulignée dans le texte du menu. Enfin, vous pouvez définir vos propres raccourcis clavier pour les actions du menu, si vous avez autorisé Utiliser les raccourcis clavier dynamiques dans le dialogue des préférences.
- 3. Bouton Menu : Cliquer sur ce petit bouton déroule le Menu d'image sur une colonne au lieu d'une ligne. Ceux qui utilisent les accélérateurs claviers et qui et qui n'ont pas la barre des menus visible peuvent accéder au menu d'image par la touche Maj F10.
- 4. Les règles : Dans la configuration par défaut, les règles sont affichées en haut et à gauche de l'image, elles indiquent les coordonnées dans l'image. Vous pouvez choisir le type de coordonnées affiché. Par défaut, l'unité de mesure est le pixel, mais vous pouvez choisir une autre unité en utilisant les paramètres d'unité décrits plus loin.

Un des usages majeur des règles est de créer des guides. Si vous cliquez sur une règle et que vous la glissez dans l'image affichée, un guide va être créé que vous pouvez dès lors utiliser pour vous positionner précisément dans l'image. Les guides sont déplaçables en cliquant dessus; ramenez les vers leur règle d'origine pour les ôter de l'image.

- 5. (Dés)activer le masquage rapide : En bas à gauche de la fenêtre image se trouve un petit bouton qui permet d'activer ou de désactiver le masque rapide qui est une manière alternative, et souvent très utile, de voir la partie sélectionnée d'une image.
- 6. Coordonnées du pointeur : Dans le coin inférieur gauche de la fenêtre image s'affiche les coordonnées du pointeur (c-à-d la position de la souris, si vous utilisez une souris), tant que le pointeur se trouve sur l'image. L'unité est la même que pour les règles.
- 7. Menu des unités : (fonctionnalité apparue dans GIMP 2.2) Par défaut, l'unité utilisée pour les règles et d'autres fonctions est le pixel. Vous pouvez la changer pour le centimètre, le pouce ou bien d'autres possibilités en utilisant ce menu (Si vous le faites, sachez que le paramètre "Point pour point" du menu Affichage modifiera les dimensions d'affichage de l'écran.
- 8. Bouton Zoom : (fonctionnalité apparue dans GIMP 2.2) Il y a plusieurs manières de zoomer plus ou moins dans l'image, mais ce menu est rapide d'accès et affiche la valeur courante du zoom.
- 9. Barre de statut : La barre de statut se trouve en bas de la fenêtre image. La plupart du temps, par défaut, elle affiche la partie de l'image actuellement active, et l'espace mémoire que l'image occupe. Vous pouvez personnaliser ces informations en changeant vos Préférences. Lorsque vous lancez des opérations consommatrice de temps, la barre de statut affiche quelle est l'opération en cours et son état d'avancement.
- 10. Bouton Annuler : Dans le coin en bas à droite de la fenêtre image se trouve le bouton Annuler. Si vous commencez une opération complexe et consommatrice de temps (la plupart du temps un greffon), et décidez, alors qu'elle est en cours, que vous ne voulez plus vraiment l'effectuer, ce bouton l'arrêtera immédiatement.
- 11. Bouton Navigation : C'est un petit bouton avec une croix dont les bras sont des flèches en bas à droite de la fenêtre image. Cliquer dessus et maintenir le bouton gauche de la souris enfoncé fait apparaître une fenêtre montrant une vue miniature de l'image et la partie actuellement affichée encadrée. Vous pouvez vous déplacer dans l'image en bougeant la souris tout en maintenant le bouton pressé. Pour les images de grande taille, dont une petite partie seulement est affichée, la fenêtre de navigation est souvent la manière la plus simple d'afficher la partie qui vous intéresse (voir dialogue de Navigation pour les autres manières d'accèder à la fenêtre de navigation). (Si votre souris possède une molette, le fait de cliquer avec dans l'image vous permet de vous déplacer dans l'image).
- 12. Partie inactive de la fenêtre image : Ce remplissage sépare, dans la fenêtre image, la partie active de l'image de la partie inactive, ainsi vous pouvez faire la distinction entre elles. Vous ne pouvez pas appliquer de filtres ni, plus généralement, faire d'opération sur la partie inactive.

- 13. Affichage de l'image : La partie la plus importante de la fenêtre image est, bien entendu, l'image elle-même. Elle occupe la partie centrale de la fenêtre, entourée d'un tireté jaune et noir montrant les limites de l'image, sur un fond gris neutre. Vous pouvez changer le niveau de zoom de la fenêtre image de diverses manières, y compris avec le paramètre du comportement du zoom décrit plus bas.
- 14. (Dés)activer le redimensionnement de la fenêtre image : Si ce bouton est pressé, l'image est redimensionnée si la fenêtre image l'est aussi.

ت الله من الله من الله من الله م	ayer 104.jpg	s, Cl 1-2	nanne	ls, –	Auto
Layers Mode:	Soft	S light		\$]	• •
Opacity:			New Back	Layer Groun	00.0 (‡ d
	<u>ه </u>	Ŷ	ل ل	Y	

2.5 Dialogues, fenêtres et onglets

FIG. 2.4 – Une fenêtre avec sa barre de rattachement

Depuis GIMP 2.0 vous disposez d'une plus grande souplesse pour organiser votre espace de travail. Au lieu d'ouvrir une fenêtre pour chaque dialogue vous avez maintenant la possibilité de les regrouper sous forme d'onglets dans une même fenêtre. Toutefois si vous pouvez regrouper les dialogues il n'en va pas de même pour les images qui restent individuellement séparées dans leur propre fenêtre. Par contre vous pouvez rattacher des dialogues considérés comme temporaires comme le dialogue des Préférences ou celui d'une nouvelle image. Chaque fenêtre dispose d'une barre de rattachement comme celle montrée sur l'image de la figure 2.4. Suivant le style de décoration de vos fenêtres elle peut être peu visible et beaucoup d'utilisateurs n'ont pas fait attention à elle au début.

✓ Cal	ques, Canaux, Chen 🗕 🗆 🗙
3D.pr	ng-1 🔽 Auto
88	(x)
Calques	I R
Mode :	Normal 🗘 🗖 🔛
Opacité :	100.0
0	Arrière-plan
۲	Fond

FIG. 2.5 – Un dialogue dans une fenêtre avec sa zone de rattachement

Chaque dialogue dans une fenêtre a une zone de rattachement (voir figure 2.5. Vous pouvez aisément l'identifier par le fait que le curseur prend l'aspect d'une main avec l'index tendu quand il se trouve au-dessus d'une zone de rattachement. Pour rattacher ici un dialogue il suffit de cliquer dessus puis de le glisser et déposer dans cette zone.

Vous pouvez rattacher plus qu'un dialogue dans la même zone de rattachement. À chaque fois un nouvel onglet sera créé. Si vous cliquez dans la zone de rattachement d'un onglet alors vous pouvez le glisser et le déposer soit sur le bureau soit sur une autre zone de rattachement dans la même fenêtre ou dans une autre fenêtre.

Certaines fenêtres de rattachemnt contiennent un menu des Images (voir la figure 2.6) qui liste toutes les images ouvertes dans GIMP et affiche leurs noms (ne confondez pas ce Menu des images avec le Menu d'image, qui est le menu de l'image active). Vous pouvez utiliser ce menu pour activer une image différente de celle qui actuellement activée. Si le bouton Auto est coché alors ce menu montre toujours le nom de l'image activée, c-à-d celle sur laquelle vous travaillez.

Par défaut, en haut d'une fenêtre d'onglets de type Calques, Canaux et Chemins on trouve un Menu des images, alors que dans d'autres non. Vous pouvez toujours ajouter ou retrancher ce Menu des Images en utilisant Afficher l'image sélectionnée (Afficher le Menu des images) dans le menu des onglets. Cette option n'est pas

✓ Cale	ques, Canaux, Chen 🗕 🗆 🗙
🚳 3D.pn	g-1 🛛 🔽 Auto
8 5	<u>ايم</u>
Calques	I X
Mode :	Normal 🗘 🗆 🖾
Opacité :	100,0
0	Arrière-plan
1	Fond

FIG. 2.6 – Un dialogue avec son menu des images

disponible quand les onglets sont rattachés sous la boîte à outils.

Chapitre 3

Utilisation de Multipass

Le greffon développé dans le cadre de ce projet se compose de deux programmes distincts correspondant aux deux modes opératoires disponibles :

- Mode image unique : Le principe est d'appliquer les algorithmes de segmentation à une image à la fois afin de mieux examiner et ajuster les résultats pour une image particulière. L'objectif étant dans ce cas d'obtenir les meilleurs résultats sur une image donnée.
- Mode « batch » : Dans le « mode batch » il s'agira de choisir une combinaison particulière d'algorithmes, d'ajuster leurs paramètres puis de les appliquer à un grand nombre d'images avant de récupérer un résumé des résultats. L'objectif étant dans ce cas de tester le comportement des algorithmes sur une grande quantité de données.

Ces deux modes sont en fait complémentaires : on pourrait utiliser le mode image unique pour ajuster au mieux les paramètres d'un algorithme avant de tester l'efficacité de ce paramétrage sur un grand nombre d'images.

Les sections suivantes constituent un manuel d'utilisation : elles détaillent l'utilisation de l'interface ainsi que l'interprétation des résultats de segmentations dans les deux modes opératoires.

3.1 Mode Image Unique

Le mode image unique opère sur une image particulière qui doit donc être ouverte au préalable dans GIMP. L'installation du greffon provoque l'ajout d'un menu « Segmentation » dans toutes les images ouvertes. C'est ce menu qui invoque le mode image unique de l'application (voir figure 3.1).

Les algorithmes de segmentation considérés ici opèrent sur des images en niveaux de gris. Si l'image traitée est une image couleur (RGB, ou couleurs indexées) la conversion en niveaux de gris se fera automatiquement. Cependant, cette conversion automatique ne donne pas toujours les meilleurs résultats et il est donc recommandé d'effectuer l'opération de conversion en utilisant les outils disponibles dans GIMP.

Une fois le menu sélectionné l'interface principale du mode image unique apparaît (Voir figure 3.2).



FIG. 3.1 – Le menu segmentation déclenche le mode image unique.

L'interface du mode image unique permet la sélection des algorithmes de segmentation à appliquer ainsi que l'ajustement des paramètres. La fenêtre se compose de plusieurs panneaux remplissant chacun un rôle prédéfini.

Le panneau « Algorithm » (en haut à droite) est le panneau principal de l'interface. Il permet de sélectionner la combinaison particulière d'algorithmes à appliquer. Il s'agit en fait d'une liste d'algorithmes qui seront appliqués de haut en bas, successivement. Un bouton « Add » positionné en bas de la liste permet d'ajouter un algorithme à la fin de la séquence.

Chaque algorithme de la liste est représenté par un sous-panneau comprenant le nom de l'algorithme, trois boutons : « Up », « Down » et « Remove », ainsi que la mention « Parameters ». Il est possible de changer à tout moment l'algorithme en question en cliquant sur son nom. Une liste de choix des algorithmes de segmentation disponibles est affichée.

En fonction de l'algorithme sélectionné dans un sous panneau, plusieurs paramètres pourront être ajustés. Ces paramètres sont, par défaut, masqués sous la mention « Parameters » dans le sous panneau. Pour les afficher (et ainsi pouvoir les ajuster), il suffit de cliquer sur le bouton « Parameters » (Voir figure 3.3).

Le deuxième panneau, à gauche de la fenêtre (titré « Preview ») permet d'obtenir un aperçu des résultats de la séquence d'algorithmes sélectionnée sur une portion de l'image.

La fenêtre d'aperçu elle-même montre une portion de l'image à laquelle a été

v	Multipass Segmentation	×
Preview	Algorithm AiNet Image: Description of the second	=
Performance Turi (2001) 0.721276		•
Melp	🗶 Cancel	

FIG. 3.2 – L'interface principale du mode image unique.

w	Multipass Segmentation 🛛 🗙
Preview	Algorithm Algorithm AiNet Down Remove Remove Fuzzy C-Means Lup Down Remove
	Pixel Classifier
Melp	Cancel

FIG. 3.3 – Ajustement des paramètres

appliquée la segmentation. L'image est recolorée pour mettre en valeur les différentes classes de pixels (les pixels de la même classe reçoivent la même couleur).

L'aperçu est généré de façon continue au fur et à mesure que la séquence d'algorithmes est modifiée et que les paramètres sont ajustés. Si la surcharge induite est trop importante, il est possible de désactiver la mise à jour de l'aperçu en cliquant sur le bouton à deux états « Preview ».

En bas du panneau « Preview » se trouve un sous panneau « Performance », qui donne une estimation de la qualité de la segmentation sur la portion de limage en cours de traitement (à savoir l'aperçu). Cette estimation est en fait le résultat de l'application d'une des fonctions de fitness disponibles. Il est possible de choisir la fonction de fitness désirée en cliquant sur la liste déroulante affichant la sélection en cours. L'aperçu ainsi que l'estimation des résultats ne concernent qu'une portion de l'image. Le résultat de l'application de l'algorithme à l'image entière peut d'être différent (qualité supérieure ou moindre).

Si l'image est enregistrée sur le disque sous un nom tel que « fichier.ext » et qu'une image de même taille est enregistrée dans le même répertoire sous le nom de « fichier_seg.ext », alors cette dernière est considérée comme une segmentation de référence pour l'image en cours. Dans ce cas, la liste des fonctions de fitness comporte une entrée supplémentaire : « Jaccard », qui donne une estimation supervisée de la segmentation. Cette convention de nommage de la segmentation de référence reste valide pour la fenêtre des résultats, ainsi que dans le mode batch.

Le dernier panneau (en bas de la fenêtre) permet de choisir entre deux méthodes d'affectation de pixels aux classes :

- La méthode classique, ou chaque pixel est affecté à la classe dont le centre est le plus proche;
- La méthode basée sur la position où le voisinage de chaque pixel influence son affectation.

Après avoir choisi les algorithmes, ajusté les paramètres, et vérifié l'aperçu, il ne reste plus qu'a cliquer sur « OK » pour démarrer la segmentation elle-même.

Compte tenu de l'implémentation adoptée, l'opération ne devrait pas durer plus de quelques secondes même pour de grandes images (1024x768 pixels, ou même 1600x1200 pixels).

Une fois la segmentation terminée, deux fenêtres apparaissent :

- L'image segmentée : qui est une copie de l'image originale dont les pixels ont été recolorés pour refléter leur classification.
- un résumé de statistiques sur la classification : Les données sont regroupées à chaque étape de la segmentation et sont présentées séparément. La fenêtre contiendra donc un tableau de statistiques résumant l'état des classes après chaque étape.

Par défaut, seuls les résultats finaux (correspondant à la dernière étape) sont affichés, les autres sont masqués. Pour les afficher, il suffit de cliquer sur la mention « Results » correspondante.

Pour chaque étape, la fenêtre affiche le nombre de classes ainsi que pour chaque classe des informations telles que :

- La couleur (qui permet de retrouver la classe dans l'image segmentée);
- La valeur du centroid correspondant;
- Le pourcentage de pixels appartenant à la classe (« Pixel Coverage Ratio »);
- La moyenne (« Mean »);
- La variance (« Variance ») et L'écart type des niveaux de gris des pixels dans la classe.

La fenêtre affiche aussi une estimation de la qualité de la classification telle que fournit par une fonction de fitness. Il est possible de choisir la fonction de fitness en question (parmi celles disponibles) en cliquant sur la liste déroulante en haut de la fenêtre.



FIG. 3.4 – Image après segmentation.

3.2 Mode «Batch»

Le mode batch n'opère pas sur une image en particulier mais sur un grand nombre d'images à la fois. C'est pourquoi le sous menu invoquant cette fonctionnalité n'est pas dans le menu « Image » de GIMP mais dans le menu principal (au dessus de la boite à outils, voir la figure 3.6).

Une fois ce menu invoqué, la fenêtre de la figure 3.7 apparaît. L'interface du mode batch est composée de deux parties :

- la partie supérieure qui représente Le panneau « Algorithm ». Ce dernier est similaire à celui présenté précédemment, il permet d'éditer la liste des algorithmes à appliquer et d'ajuster leurs paramètres ;
- la partie supérieure qui représente Le panneau « Algorithm ». Ce dernier est similaire à celui présenté précédemment, il permet d'éditer la liste des algorithmes à appliquer et d'ajuster leurs paramètres.

La dernière étape consiste à choisir un fichier où seront sauvegardés les résultats, soit en entrant directement *son nom* dans la zone « Save results to », ou en parcourant les disques durs à la recherche d'un emplacement en cliquant sur le bouton «...».

Les résultats d'une segmentation en mode batch consistent en un fichier texte résumant des données statistiques correspondantes à l'opération dans son ensemble.

Pour chaque image, le fichier contiendra le nombre de classes résultantes, le

After AiN	et ults : of classes: 6					
Colors	Centers	Pixel Coverage Ratio (%)	Mean	Variance	Root Variance	
	153.000000	3.877936	148.809353	207.867902	14.417625	
	204.000000	1.534435	198.364069	159.369185	12.624151	
	255.000000	0.092996	238.314286	45.729796	6.762381	
	102.000000	13.205442	96.319215	196.653333	14.023314	
	51.000000	33.028218	49.485700	185.759791	13.629372	
	0.000000	48.260974	6.945481	33.035167	5.747623	
Perform	nance:	1.055				

FIG. 3.5 – Données statistique sur la segmentation.

temps d'exécution, ainsi que le résultat de l'application des fonctions de fitness. Ce fichier pourra alors aisément être introduit comme entrée pour un tableur à partir duquel pourront être produits des graphes permettant une lecture claire et précise des résultats obtenus et des comparaisons de résultats de plusieurs applications.

*	The GIMP		6
<u>F</u> ile	<u>X</u> tns <u>H</u> elp Se	gmentation	
	* P 🔨	Iltipass Segme	ntation (Batch Mode)
P	P Å +	1 🕸 🖪	
<u>,</u>	🚨 🛃 T 🤇	3 🗖 🧷	
1	- 2 I	8 6 6	
đ			
	\$		
-			
aintb	orush	3 8	
Opac	ity:	100.0	
Mode	Normal	+	
Brush	n: • Circle (11)	
D Pr	essure sensitivit	y	
🗌 Fa	de out		
	cremental	•	
		ൈ	

FIG. 3.6 – Menu du mode "batch"

AiNet		\$		Down	emove <u>R</u> emove	
	neters					
Number o	f Iterations	i 📖 💻				-
Sigma_S	30	D				
Sigma_D	_10					
N 3 [0.200					
Zeta (%)	0.300					
Particle	Swarm Optim	lization 😂	₫ <u>U</u> p	• <u>D</u> own	- <u>R</u> emove	
hmark			Nononononi .			
<i>i</i> a raculte	to:					
> <u>A</u> dd	- <u>R</u> emove	Save	Den Open			
/home/a	nis/Desktop/in	nages/non-super	vised/0003.b	mp		
/home/a	nis/Desktop/in	nages/non-super	vised/0004.b	mp		
/home/a	nis/Desktop/in	nages/non-super	vised/0005.b	mp		
11	nie/Deekton/in	ages/non-super	vised/0006 h	mn		

FIG. 3.7 – Interface du mode batch.

Chapitre 4

Extension de Multipass

Les paragraphes qui suivent détaillent le schéma de programmation suivi par les algorithmes. Ils décrivent donc les étapes à suivre pour étendre l'application et ajouter de nouveaux algorithmes.

Dans le cadre de ce travail, un algorithme de segmentation particulier dans le contexte d'un enchaînement d'algorithmes est appelé une passe. Mise à part une solution initiale et un histogramme, une passe aura souvent besoin de plusieurs autres paramètres permettant de personnaliser son comportement (par exemple, un seuil de précision, ou le nombre de classes initiales). Ces paramètres ne sont pas universels et dépendent de la passe particulière considérée. Il relève donc de la responsabilité de la passe elle-même de déclarer le nombre et le type de paramètres qu'elle requière pour que l'application puisse générer l'interface utilisateur permettant d'ajuster ces paramètres, et ainsi permettre à la passe elle-même de consulter les valeurs choisies par l'utilisateur.

Afin d'illustrer cette notion, nous décrirons une passe modèle, qui contient tout le code nécessaire excepté l'algorithme de classification lui-même. Cette passe modèle se trouve implémentée dans les fichiers template_pass.hpp et template_pass.cpp. Ces fichiers peuvent donc servir de point de départ pour le développement de nouveaux algorithmes dans l'application.

Noter que dans ce paragraphe nous aborderons certaines classes et méthodes faisant partie du noyau de l'application. Cependant, ce document n'étant en aucun cas un substitut de la documentation présente dans le code source lui-même, il est recommandé au lecteur de se référer au code source et principalement aux fichiers d'entête pour plus de détails.

4.1 Le fichier d'entête

Le seul objet de template_pass.hpp est de déclarer la classe template_pass :

```
/// The number of parameters we need
enum { m_param_info_size=2 };
/// Holds a param_info for each parameter we take
static param_info m_param_info[m_param_info_size];
```

Le fichier commence par déclarer la classe template_pass comme une classe fille de la classe pass. Il est aisé de déduire que la classe pass (c.f. fichier pass.hpp) est la parente de toutes les classes représentants une passe.

L'étape suivante est d'informer la classe **pass** du nombre et du type de paramètres que **template_pass** requière. Ceci est obtenu en constituant un tableau d'objets de type **param_info**. Un objet de type **param_info** (c.f. **param_info.hpp**) contient des informations sur un paramètre d'une passe. Le programme ci-dessus déclare simplement un tableau statique (**m_param_info**) d'objets **param_info** appelé à contenir l'information sur les paramètres de la passe. Nous noterons que ce code porte uniquement sur la déclaration du tableau qui sera rempli à une étape ultérieure.

public:

}

Puis vient l'interface public de la classe. Il est à noter que le constructeur de la classe **pass** prend trois arguments :

- 1. Une étiquette (char const*) : Qui servira de nom externe à la passe, et permettra à l'utilisateur de l'identifier parmi les autres. Ce nom doit être unique dans le contexte des autres passes de l'application;
- Un tableau d'objets de type param_info : Qui contient un objet param_info pour chaque paramètre de la passe. Noter comment le code passe le tableau m_param_info vue précédemment ;
- 3. Le nombre d'éléments dans le tableau.

Le code déclare ensuite le membre le plus important. C'est cette surcharge de « l'opérateur parenthèses » (operator()) qui va effectuer la segmentation elle-même. Nous constatons aisément que les deux premiers arguments ne sont autres que les deux données en entrée d'un algorithme que nous avons décrites précédemment, à savoir : L'histogramme (c.f. histogram.hpp) et un ensemble de centroids (c.f. centroid_set.hpp).

Il est également très important de noter que l'histogramme est passé en tant que référence constante, la passe ne peut donc pas le modifier. Cependant, l'ensemble des centroids est passé, lui, en tant que référence non constante. Ce qui signifie qu'il est un argument en entrée (il donne la solution initiale), mais aussi un paramètre en sortie, car il permet à l'algorithme de retourner la solution améliorée directement dans le paramètre lui même. En d'autres termes, l'objectif est de mettre à jour la solution reçue.

Le dernier paramètre n'a pas d'impact sur la segmentation elle-même, mais permet à la passe de renseigner l'application sur le taux d'avancement courant de l'algorithme. Cet argument possède un membre operator() que nous pouvons invoquer en passant une estimation du pourcentage de la progression de l'algorithme (0.0 pour aucun travail effectué pour l'instant, 1.0 pour travail terminé). (voir le fichier progress_indicator.hpp pour plus de détails).

```
/// Must return a new copy of *this
virtual pass* clone() const { return new template_pass(*this); }
;
#endif // TEMPLATE_PASS_HPP
```

La derrière partie du fichier contient le dernier membre de la classe : clone(), dont le rôle est de retourner une copie de l'instance courante allouée avec l'operateur new.

4.2 Le fichier source

Le rôle du fichier source est de définir les paramètres de la passe et l'algorithme lui-même :

```
#include "template_pass.hpp"
#include "histogram.hpp"
/// Definition of parameter info array
///(contains an element per parameter)
param_info
template_pass::m_param_info[template_pass::m_param_info_size]=
{
    /// First parameter will be an integer from 1 to 10 named
    /// "Number of Iterations"
    param_info(1, 10, "Number of Iterations",
                    "Number of passes to apply"),
    /// Second parameter will be a double between 0 and 1
```

```
};
```

Γ

Nous notons la définition du tableau m_param_info mentionné précédemment et comment le programme le remplit avec des instances de *param_info* (voir *param_info.hpp*).

Enfin vient l'implémentation de l'algorithme lui-même. La première opération à effectuer est de récupérer les valeurs des paramètres choisis par l'utilisateur. D'après ce qui précède, la passe prend deux arguments : un entier entre 1 et 10, et un réel entre 0.0 et 1.0.

Pour récupérer les valeurs effectives de ces paramètres il suffit d'invoquer la fonction pass::parameter_value() (implémentée dans la classe parente : pass) en passant l'index du paramètre concerné. La valeur retournée par parameter_value() ne peut être simplement un int ou un double, du fait que l'information sur le type du paramètre ne peut être connue au moment de la compilation. C'est pourquoi la valeur de retour est un objet de type param qui est un conteneur de données flexible contenant soit un entier soit un réel. Il reste simplement à invoquer param::int_value() pour obtenir un réel (dans le cas de paramètres réels).

}// while pass_count--

}

Le reste du fichier est un modèle vide auquel il faudra ajouter un algorithme de segmentation. Dans un exemple réel, le code utilisera l'histogramme et la solution initiale pour améliorer cette dernière.

Après avoir procédé à l'écriture du programme lui-même, il faut l'enregistrer auprès de l'application pour qu'il apparaisse dans la liste des algorithmes disponibles. Pour ce faire, nous devons procéder à l'ouverture du fichier **register_pass.cpp** et apporter des modifications comme indiqué dans le fragment ci-après :

```
#include "register_passes.hpp"
#include "pass_repertoire.hpp"
#include "ainet_pass.hpp"
#include "kmeans_pass.hpp" // <<< See here ///////
void register_passes()
{
    pass_repertoire& rep=*pass_repertoire::instance();
    rep.add(new ainet_pass());
    rep.add(new kmeans_pass());
    rep.add(new template_pass()); // <<< See here ////////
}</pre>
```

La dernière étape porte sur la recompilation de l'application sans omettre d'y inclure les fichiers nouvellement créés.

Chapitre 5

Compilation et Installation de Multipass

Les paragraphes suivants détaillent la procedure a suivre pour compiler et installer Multipass afin qu'il devienne accessible depuis les menus de GIMP.

5.1 Compilation

Avant de pouvoir compiler l'application il faut installer un certain nombre de bibliothèques dont l'application dépends. Ces bibliothèques sont :

libgimp : C'est l'API de GIMP lui même ;

libgtk : C'est la bibliothèque d'interface utilisateur sur laquelle repose GIMP et Multipass;

libglib : Une bibliothèque de gestion de « threads » (entre autres).

Il sera nécessaire d'installer également toutes les dépendances des bibliothèques elles mêmes avant de pouvoir procéder à la compilation.

Sous le système Debian GNU/Linux 4.0 («etch») la procédure complète d'installation des dépendances est donnée ci-dessous :

root@thunder:~# aptitude install libgimp2.0-dev libgimp2.0-doc

Le système de paquets gérant automatiquement les dépendances l'installation de libgimp2.0-dev entraînera l'installation de libgtk2.0-dev ainsi que de libglib-dev.

Une fois l'installation des dépendances effectuée la compilation de Multipass consiste simplement a invoquer la commande **make** comme indiquer ci dessous :

```
anis@thunder:~$ cd multipass
anis@thunder:~/multipass$ ls
AUTHORS COPYING doc README src
anis@thunder:~/multipass$ cd src
anis@thunder:~/multipass/src$ make
anis@thunder:~/multipass/src$ ls -1 multipass multipass_batch
-rwxr-xr-x 1 anis anis 158072 2007-09-07 14:03 multipass
-rwxr-xr-x 1 anis anis 157935 2007-09-07 14:03 multipass_batch
anis@thunder:~/multipass/src$
```

La commande make créera deux fichier exécutable : multipass et multipass_batch qui correspondent aux deux modes opératoires de l'application (image unique et batch).

5.2 Installation

L'installation consiste simplement a copier les deux fichier exécutables produis durant la phase de compilation dans le répertoire des greffons de GIMP, pour que ce dernier puisse «trouver» Multipass.

Le répertoire d'installation des greffons varie selon le système d'exploitations considéré. Dans le cas des systèmes «UNIX» (tel LINUX, ou les systèmes BSD) il s'agit de ~/.gimp2.2/plug-ins/. La procédure d'installation est donc simplement :

```
anis@thunder:~$ cd multipass
anis@thunder:~/multipass$ cd src
# Installation
anis@thunder:~/multipass/src$ cp multipass multipass_batch \
~/gimp2.2/plug-ins
# Ou bien
```

```
anis@thunder:~/multipass/src$ make install
```