Propagation de Marquages pour le Matting Vidéo

Marwen Nouri

Thèse

Soumise en en vue de l'obtention

du grade de

Docteur de l'Université Paris Descartes

UFR Mathématiques et Informatique

Propagation de Marquages pour le Matting Vidéo

Par

Marwen Nouri

2013

Direction de thèse : Pr. Nicole Vincent

Soutenue publiquement le 31 Janvier 2013 devant le jury suivant:

Pr	Sébastien	LEFEVRE	Université de Bretagne Sud	Rapporteur
Pr	Serge	MIGUET	Université Lumière Lyon 2	Rapporteur
Pr	Vincent	CHARVILLAT	IRIT UMR CNRS	Examinateur
Dr	Emmanuel	MARILLY	Alcatel-Lucent Bell Labs	Examinateur
M	Olivier	MARTINOT	Alcatel-Lucent Bell Labs	Invité
Pr	Georges	STAMON	Université Paris Descartes	Examinateur
Pr	Nicole	VINCENT	Université Paris Descartes	Directeur

بسم الله الرحمان الرحيم

قال الفضيل بن عياض: "لا يزال العالم جاهلا بما علم حتى يعمل به, فإذا عمل به كان عالما"

إلى ميسان وياسين

إلى والديا و إخوتي

إلى عفاف

« Le savant restera ignorant de ce qu'il a appris jusqu'à ce qu'il œuvre sur ceci. Ainsi une fois qu'il aura œuvré, il deviendra savant » (Fadhayl ibn Iyad)

A Mayssane (maysouna) et Yacine (yacino)
A mes parents et mes frères
A Afef

Remerciements

En premier lieu, je souhaite exprimer au professeur Nicole Vincent, mon directeur de thèse, l'expression de mes chaleureux remerciements pour sa disponibilité au cours des moments clés et difficiles de cette thèse ainsi que pour les élans qu'elle a su donner à mes recherches tant lors des phases de recherche, des haut des bas, etc. Ces travaux lui doivent beaucoup et moi aussi.

J'adresse à Olivier Martinot et Emmanuel Marilly, mes encadrants au sein Bell Labs d'Alcatel-Lucent, mes sincères remerciements, toute ma reconnaissance et mon amitié. Par l'occasion, je tiens à remercier tous les collègues des Bell Labs, qui m'ont soutenu depuis mon arrivée en 2009 en stage de fin d'étude jusqu'à aujourd'hui.

Un Grand merci également, à tous les membres de mon équipe, SIP, les anciens, les nouveaux, tous m'ont soutenu et aidé à surmonter les moments les plus difficiles. Merci à vous.

Merci également à tous les membres du jury qui m'ont fait l'honneur de prendre part et de participer à cette thèse.

D'autres personnes ont participé à l'aboutissement de ces travaux de recherche. Je prie, par avance, les personnes qui ne sont pas citées ici, comme il se devrait, de bien vouloir m'excuser.

Je vous dis un grand merci à vous, mes parents. Je te dis un grand merci à toi, ma femme. Un grand merci vous les ouafi. Finalement, un merci pour mes petits cœurs qui ont rempli m vie de bonheur.

Un grand merci à toute ma famille, mes amis, à tous ceux qui étaient présent pour moi, je vous dis merci.

Résumé

Cette thèse porte sur l'élaboration d'un système de manipulation de vidéo. De manière plus précise il s'agit d'extraction et de composition d'objets vidéo. Dans le domaine du traitement d'image fixe, les techniques d'extraction et de démélange (connus sous le nom de *matting*) et de composition ont vu une réelle amélioration au cours de la dernière décennie, surtout avec l'apparition de méthodes semi-automatiques profitant d'une interaction avec l'utilisateur pour surmonter le gap sémantique. Cela a permis d'aboutir à des algorithmes de plus en plus rapides et de plus en plus robustes. Dans le cadre du traitement de vidéo, cette problématique forme encore un très intéressant challenge, issu du caractère volumineux, en termes complexité de données et de nombre d'images dans la vidéo. Cet élément fait en sorte que la tâche accomplie par l'utilisateur pour marquer un objet d'intérêt peut être très fastidieuse ou souvent impossible. Les travaux que nous avons réalisés au cours de cette thèse se sont concentrés sur l'extension et l'adaptation de la transformée en distance et des courbes actives pour la propagation des marquages d'objets vidéo. Nous avons aussi proposé une amélioration d'une technique pouvant être utilisée avec ces marquages pour l'extraction d'objet vidéo.

Dans le premier chapitre nous présentons le contexte et la problématique de nos travaux. Dans le deuxième chapitre nous faisons un tour d'horizon des approches, des outils d'édition de vidéo existant sur le marché, tout en les classant en deux familles : édition par morceaux ou par blocs et édition par objets vidéo. Ensuite, nous présentons un rapide état de l'art sur la segmentation que nous décomposons en trois parties : la segmentation classique, la segmentation interactive et l'*image matting*. Aussi nous détaillons l'extension de l'*image matting* au *video matting* en présentant les principales approches existantes. Le chapitre 3 présente notre première approche pour la propagation de marquage dans les vidéos. Cette approche est une approche volumique 2D+T tirant sa puissance de ce que nous avons bâti une CDT (transformée en distance couleur). Le chapitre 4, lui, présente notre évolution de perception vers un processus de propagation de marquages plus robuste et plus performant basé sur les courbes actives. Nous commençons par faire un état de l'art abrégé sur les courbes actives et nous présentons par la suite notre modélisation et son application. Nous détaillons, aussi le mécanisme de gestion dynamique des poids que nous avons mis en place. Dans le chapitre 5, nous allons discuter de l'application de notre système pour le *matting* vidéo et nous présentons les améliorations que nous avons apportés à l'approche *Spectral Matting*, dans ce but.

TABLE DES MATIERES

FIGURES

TABLEAUX

ALGORITHMES

CHAPITRE 1. INTRODUCTION

1.1 CADRE INDUSTRIEL ET CONTEXTE DE L'ETUDE

Dans ce chapitre, nous allons définir le contexte industriel et scientifique dans lequel ont été réalisés nos travaux.

1.1.1. CADRE INDUSTRIEL

Cette thèse est une thèse CIFRE qui s'est déroulée en collaboration avec les Bell Labs d'Alcatel-Lucent. Alcatel-Lucent axe ses activités autour d'un certain nombre d'activités innovantes et cela se fait en partie grâce à ses choix et activités de recherche au sein des Bell Labs. Différentes directions y sont explorées, en partant du réseau, de la physique, des mathématiques, etc., jusqu'au multimédia. Dans le domaine du multimédia, les activités se concentrent particulièrement sur la vidéo qui constitue 90% des données qui transitent sur Internet et donc une grande majorité des données transitant sur les composants d'infrastructure vendus par l'équipementier. Ces éléments rendent la recherche en multimédia un élément primordial aux Bell Labs pour, soit améliorer le traitement des données transitant sur le réseau, soit pour faire naître ou créer de nouveaux usages permettant de valoriser les technologies vendues à travers ses équipements.

1.1.2. CONTEXTE DE L'ETUDE

Au cours des dernières années nous avons vu apparaître de nouveaux usages qui ont contribué à la transformation du monde de l'informatique. Les évolutions et les progrès technologiques récents ont donné lieu à de nouvelles applications orientées de plus en plus vers une implication de l'utilisateur, ce dernier étant sollicité au cours d'un processus interactif. Il apporte des connaissances d'une grande utilité et des informations non disponibles dans les systèmes. Il peut aussi apporter une assistance au système quand ce dernier ne trouve pas de solution, ainsi le système peut apprendre pour pouvoir, dans le futur, être plus performant. Les applications s'enrichissent et s'inspirent des technologies web 2.0

en pleine expansion où le rôle de l'utilisateur évolue pour devenir acteur majeur dans la production. En effet, nous sommes témoins d'une explosion des contenus partagés au sein de la communauté de l'internet. Cette expansion de la quantité de données rend nécessaire le développement d'outils automatiques, qui permettraient de gérer de très grandes quantités de données. L'avantage par rapport à un processus manuel est simplement de rendre les opérations faisables ou non

Actuellement, les systèmes existants n'ont pas toujours la fiabilité nécessaire pour accomplir des modifications avancées sur les contenus. La recherche de robustesse des traitements agit directement sur le rôle donné aux utilisateurs, ces derniers ne sont alors plus uniquement des consommateurs passifs, ils participent pour enrichir les systèmes et adapter les contenus à leurs envies. C'est donc la naissance d'une nouvelle ère, une fin de la séparation des concepts et de l'architecture serveur/client classique : producteur/consommateur. Ce nouveau modèle se caractérise par le fait que l'utilisateur, de plus en plus, crée ou participe à la création de ce qu'il consomme. Il prend de plus en plus, à son compte, des tâches de production. Il est donc nécessaire de lui fournir des outils pour lui permettre de créer de nouveaux contenus plutôt que de lui fournir les contenus eux-mêmes. Ici, en changeant de point de vue, on peut considérer que c'est le rôle du producteur qui lui aussi change. Mais, produire des outils n'est-il pas plus intéressant que produire du contenu ? C'est à cet aspect que notre de travail de thèse va contribuer.

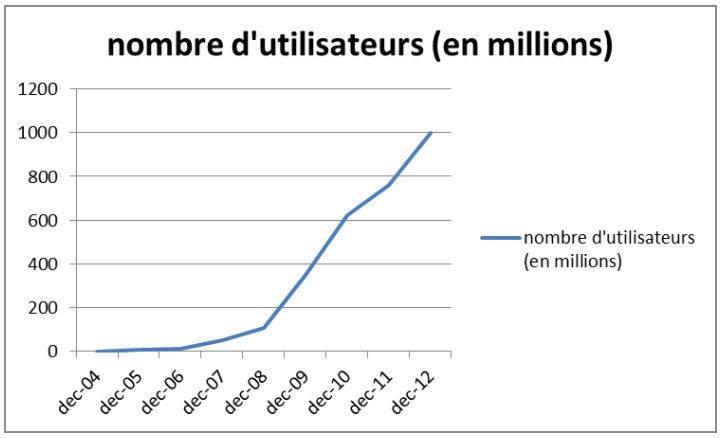


FIGURE 1 : CROISSANCE ET ADOPTION DE L'USAGE DES MEDIA SOCIAUX, ICI FACEBOOK

De nos jours, de plus en plus, presque toutes les personnes possèdent un accès internet et de plus en plus de personnes possèdent leur propre page sur la toile (plus d'un milliard d'utilisateurs sur Facebook, Figure 1). Sur cette page, ils gèrent leur profil et font figurer des contenus, tout en les partageant avec d'autres internautes (leurs amis). L'utilisateur partage de plus en plus des informations qui lui sont propres et des contenus qu'il crée ou qu'il récupère sur la toile. Son objectif est souvent de

se différencier des autres internautes, pour cela il tente aussi d'ajouter sa touche personnelle aux contenus qu'il fait figurer sur sa page personnelle afin de se l'approprier même s'il le maîtrise à peine. La nature des médias disponibles a changé, l'utilisation de la vidéo (aujourd'hui le moyen de communication le plus complet) rend la tâche de personnalisation d'un document assez complexe, surtout pour un utilisateur lambda. L'abondance actuelle, sur le web, de vidéos (120 millions, sur YouTube par exemple, avec 200000 nouvelles vidéos chaque jour, Figure 2), d'images et d'informations diverses, en plus du désir croissant du grand public de partager et de créer de nouveaux contenus en combinant plusieurs sources, nous fait sentir combien il est nécessaire de concevoir de nouveaux outils. Ceux-ci doivent être assez simples et génériques pour rendre plus aisées les manipulations de vidéos et divers médias, par le public. Il est alors nécessaire de concevoir de nouvelles tâches et de développer de nouveaux algorithmes.

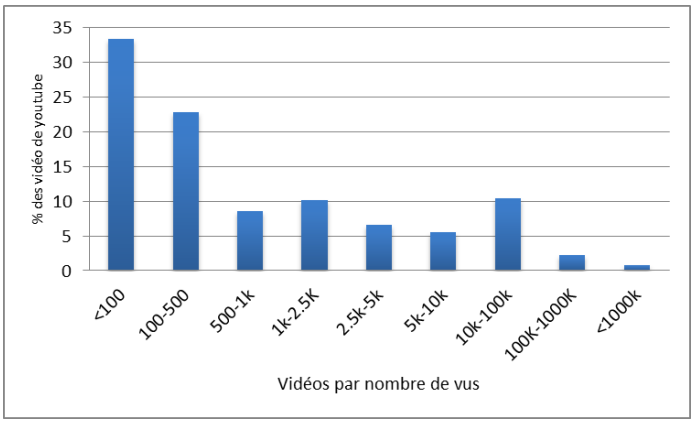


FIGURE 2 : NOMBRE DE VIDEOS VUES ET PARTAGES SUR YOUTUBE

La très grande variabilité des profils des utilisateurs, c'est à dire de leurs aptitudes techniques à manipuler, à modifier ou à créer des documents vidéo, autant que la méconnaissance que nous avons du contenu des vidéos qu'ils veulent publier, impose une étape d'extraction de connaissances qui restent élémentaires. C'est à partir d'une segmentation et d'annotations progressives que nous atteindrons un niveau de compréhension qui sera adapté aux besoins de l'utilisateur. Une vidéo est constituée de différents objets, ils peuvent être en mouvement ou statiques. Pour l'utilisateur, la notion d'objet peut varier, pour lui, l'objet d'intérêt peut être un individu, ses lunettes, sa montre ou encore l'ensemble individu plus lunettes.

1.2 PROBLEMATIQUE

Notre perspective est de réaliser une segmentation précise d'un objet offrant un rendu visuel de bonne qualité, ce qui permet d'effectuer des compositions naturelles qui passent inaperçues. Cela non pas à partir d'un mot ou une représentation sémantique mais en utilisant l'aide fournie par l'utilisateur pour désigner des centres d'intérêt à travers des annotations ou marquages. Tous les éléments de la vidéo n'ont pas besoin d'être reconnus ou analysés, l'utilisateur désigne uniquement les portions qui lui semblent les plus pertinentes. La granularité de l'information dépend alors des objectifs qui sont visés par l'utilisateur. Avant de fournir des outils facilitant l'interaction avec des vidéos, dans le but de créer de nouveaux contenus, tout en répondant aux attentes des utilisateurs, une analyse des besoins s'impose et nous avons relevé les problématiques suivantes, elles constituent autant de défis que de verrous technologiques :

1. Comment fusionner des informations de sources différentes pour pouvoir injecter de la compréhension et donc de l'intelligence et rendre plus pertinente et plus aisée la façon de manipuler les vidéos ? Il s'agit ici d'offrir à l'utilisateur différents niveaux de granularité de segmentation et d'envisager une approche interactive des algorithmes développés.

2. Comment guider les zones de recherche dans l'image selon la nature de l'image et son contexte ? Cela permettra de diminuer les bruits ou les fausses pistes, c'est une forme d'information externe à la vidéo que nous devons intégrer.

3. Disposant de tous ces éléments, comment reconstruire un objet à partir de la vidéo, non pas de manière statique, image par image mais plutôt de façon spatiotemporelle sur la séquence? Les différents descripteurs ne seront plus considérés de manière traditionnelle, mais nous nous attachons à étudier des structures 3D (2D+T).

4. Comment localiser les différents objets contenus dans la vidéo ?

5. Quelle modélisation pouvons-nous établir ?

Au cours de cette thèse nous allons examiner comment cette riche thématique de recherche s'est développée au sein de la communauté industrielle et scientifique. Suite à cette analyse, nous allons mettre en évidence un certain nombre de solutions possibles. Cela en répondant aux questions que nous nous sommes posées.

1.3 APPROCHE ET METHODOLOGIE

Dans la littérature nous pouvons retrouver plusieurs approches qui peuvent être utilisées pour accomplir la segmentation d'objets vidéo. Ces approches de segmentation peuvent être divisées en deux familles principales : les approches de segmentation automatiques et les approches de segmentation interactives. Ces dernières se décomposent elles-mêmes en deux types : segmentation binaire et segmentation continue (connue sous le nom d'*image matting*). Quelle que soit l'approche, pour atteindre une segmentation de haut niveau et dépasser le problème du gap sémantique, de manière générale des informations supplémentaires externes à l'image sont nécessaires. Ces informations peuvent être issues d'un processus d'apprentissage, dans ce cas nous nous retrouvons dans un processus global où un ensemble d'images annotées est étudié pour en extraire des informations permettant d'aller à un niveau plus fin et d'analyser les images en se basant sur un a priori. Une manière plus simple et plus générique, mais qui n'est pas tout le temps possible, consiste à demander une assistance de la part de l'utilisateur. Nous nous plaçons alors dans un processus interactif. Cette dernière façon de faire est très répondue, surtout, dans le domaine médical où le spécialiste, via une interface graphique, fournit au programme les informations et la connaissance manquante pour accomplir une tâche d'extraction d'éléments donnés. Ces informations peuvent être fournies au système sous la forme de points, de traits, d'annotations ou de marquages quelconques.

Dans le monde des applications multimédia grand public, la notion d'intégrer l'utilisateur au sein du système est arrivée un peu plus tard. Ces dernières années, nous pouvons voir une prolifération des approches semi-automatiques et interactives. En effet, cette intégration de l'utilisateur permet d'avoir des champs de compréhensions et de connaissances plus larges et permet aux applications de mieux intégrer le monde de la vidéo. Une des grandes difficultés à laquelle nous faisons face en nous positionnant comme étant développeur d'algorithmes et de solutions autour de la vidéo et du multimédia est de permettre de manière intuitive et simple à l'utilisateur de combler le manque d'information afin de résoudre des problématiques qui sont souvent complexes. Ainsi que nous l'avons dit précédemment, c'est à travers une phase interactive que nous tentons de profiter des connaissances fournies par l'utilisateur de manière explicite ou implicite pour lui offrir des solutions qui lui sont plausibles.

1.3.1 MODEL INTERACTIF : INTERACTION AVEC L'UTILISATEUR

L'interaction entre le système informatique et l'utilisateur permet d'acquérir de la connaissance que nous pouvons qualifier, dans la plupart des cas, d'informations sûres. Ces informations permettent de faire face au problème du gap sémantique. Le gap sémantique correspond au manque de corrélation entre, d'une part, les informations extraites de manière automatique à partir d'une image ou d'une vidéo, ou de données quelconques et, d'autre part, l'interprétation que quelqu'un peut avoir en analysant ces informations là dans le même contexte ou dans un contexte différent. Dès 1945, cette volonté ou plutôt cette nécessité d'intégrer l'utilisateur au cœur des systèmes informatiques est apparue. Vannevar Bush, dans son article 'As We May Think'[1] décrivait le concept d'un système permettant le stockage et la recherche d'informations. Dans ce système, il avait introduit de nouveaux concepts tels que le concept d'hypertexte : navigation, indexation, annotation. Cela fut l'une des premières modélisations de l'interaction homme machine au sens informatique. Avec l'apparition des premières interfaces graphiques IHM et les premiers outils d'interactions avancés tels que la souris (Figure 3), les notions de l'extension de la connaissance humaine vers l'ordinateur et inversement, d'ergonomie, de tâches semi-automatisées, etc., sont apparues.

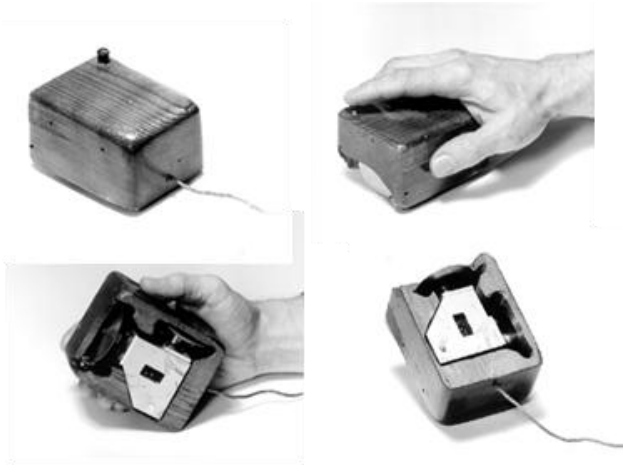


FIGURE 3 : PERMIERE SOURIS, 1964

D'autres outils et usages de la relation entre l'homme et la machine ont aussi été imaginés dans la même période, nous pouvons citer le SketchPad (1963), du fait que c'est un outil d'interaction destiné au traitement d'images.(Figure 4)

[1] http://www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/

FIGURE 4 UTILISATION DU STYLO OPTIQUE EN 1963

Pour plus de détails sur les outils et mécanismes d'interactions homme machine on peut consulter [Foo 2006].

L'usage d'un processus complètement automatisé est souvent employé à destination de cas bien spécifiques connus et maitrisés. Une approche interactive prend tout son sens dans le cadre de cette thèse où nous voulons traiter un cas générique basé sur des contenus issus du web. Ceux-ci sont très variés. Aussi, comme nous nous adressons directement à l'utilisateur final et que les résultats d'une segmentation binaire manquent parfois de finesse et donnent un aspect de découpe brute, nous avons choisi de traiter des approches de segmentation continue. Le problème du *matting* est un problème mal posé, l'interaction avec l'utilisateur est alors essentielle pour aboutir à des solutions et résoudre ce problème. Plusieurs réflexions partant de l'approche de [Ruzon and Tomasi 2000] ont modélisé une interactivité nécessaire et suffisante entre l'utilisateur et le programme pour faire du *matting*, ce modèle d'interaction est appelé *trimap*. D'autres façons assurant une bonne interaction ont été développées par la suite. Dans la section 2.4.3, nous reviendrons en détail sur ces différents éléments. Dans le cadre du *matting* vidéo, l'incorporation de la connaissance de l'utilisateur dans le système présente un enjeu plus complexe dans le cas d'images fixes. Sans cette connaissance, la solution pour résoudre un tel problème est d'intervenir au cours du processus d'acquisition par l'utilisation de matériels ou de conditions bien spécifiques. Dans notre cas, nous adressons une problématique où nous voulons rendre possible pour l'utilisateur de profiter de la grande masse de vidéos déjà disponibles sur internet. Il nous est donc impossible de prédéfinir des conditions particulières de tournage. Le problème majeur tient dans le simple fait d'utiliser une interface graphique pour

demander à l'utilisateur d'insérer des marqueurs pour définir son objet d'intérêt. La quantité de données à traiter est alors considérable. En effet marquer ou indiquer l'objet à extraire image par image est très fastidieux et une seconde de contenu vidéo contient en moyenne 25 images.

1.3.2 REPRESENTATION ET MODELISATION D'UN MARQUAGE

Un marquage fait par l'utilisateur pour désigner un objet vidéo est constitué par un ensemble de traits ou de courbes dessinés par l'utilisateur en utilisant la souris ou un dispositif de pointage tactile. Dans la littérature, pour ce type de marquage, on emploie souvent le mot '*scribbles'* (gribouillis) pour faire allusion à ce moyen d'interaction homme/machine. Un gribouillis dessiné par l'utilisateur peut être considéré sous plusieurs formes selon le contexte et la manière dont le traitement est fait. Plusieurs philosophies peuvent être considérées. Nous sommes dans le cadre ou nous disposons d'un marquage et d'un ensemble d'images. En effet, nous voulons limiter la tâche de l'utilisateur à un nombre restreint d'images. Il nous appartient alors de propager un marquage sur une image aux suivantes pendant une durée aussi longue que possible. Pour explorer cet ensemble de données, ces images, en partant des marquages faits par l'utilisateur nous pouvons considérer deux approches :

- une approche où nous considérons le marquage comme un ensemble d'éléments actifs qui explorent leur milieu (pixels et images voisins) pour trouver le chemin le plus probable qu'un marquage aurait pris d'une image à l'autre. Ces éléments creusent le contenu de la vidéo pour se créer un chemin. Dans le chapitre 3 nous allons voir la mise en œuvre de cette approche.
- Dans le chapitre 4, une autre vision est utilisée. Le marquage est considéré comme un ensemble d'éléments inertes qui sont soumis à des forces et qui se positionnent en fonction de ces forces pour atteindre un équilibre. Ces éléments se laissent aller et suivent les courants et les tensions formés par les données les entourant. Dans ce sens, ils progressent d'image en image pour former successivement les positions les plus probables des marquages dans la vidéo.

1.3.3 ARCHITECTURE

Le centre de nos recherches, ici, s'articule autour des problématiques de segmentation d'objets vidéo, ou plutôt de *matting* vidéo. Dans ce contexte nous nous plaçons dans le cas de vidéos quelconques issues de la grande bulle de l'internet. Nous devons, alors, prendre en considération que l'arrière-plan

de la vidéo puisse être dynamique et que les objets d'intérêt soient déformables et mobiles et aussi que les conditions de tournage ne soient pas maîtrisées, la caméra peut être mobile.

Après avoir étudié les approches et les contributions existant dans ce domaine et après avoir exploré les solutions commerciales existantes, pour extraire un objet vidéo (*video matting*), nous proposons une architecture en trois modules (Figure 5) basée sur un algorithme de sur-segmentation (S) comme prétraitement, prenant en entrée un marquage fourni par l'utilisateur (donc une ou plusieurs interactions I) , lisant les images de la vidéo au fur et à mesure (Li) et offrant en sortie un objet vidéo.

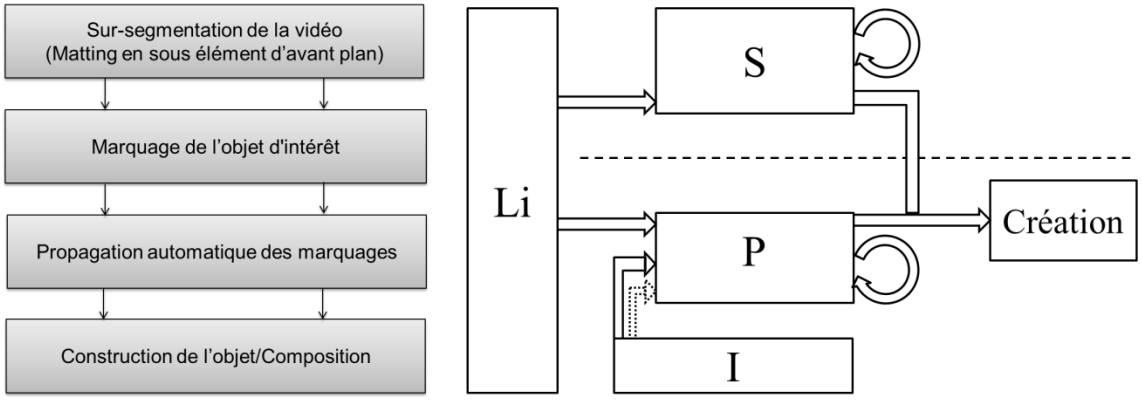


FIGURE 5 : DEMARCHE PROPOSEE POUR L'EXTRACTION D'UN OBJET VIDEO

Précisons les trois modules :

- Un module de sur-segmentation (plutôt sur-*matting*) décomposant chaque image de la vidéo en sous composants d'avant-plan. Ce module consiste à appliquer un algorithme de *matting* d'image capable de fournir plusieurs sous parties de l'image globale sous la forme d'une sur-segmentation. Chaque sous-partie est modélisée sous la forme d'un masque dont les éléments ont des valeurs allant de 0 à 1, indiquant la transparence de chaque point de la sous-partie. Cette tâche est appliquée sur chaque image de la vidéo (ou de l'ensemble des vidéos, dans le cas où nous avons une base de vidéos).

- Un deuxième module propageant, au reste de la séquence vidéo, la connaissance fournie par l'utilisateur sur la première image. Ce module fonctionne en temps réel ce qui permet à l'utilisateur de voir, au fur et à mesure des images, que son marquage continue à désigner son objet d'intérêt ou que le marquage n'est plus correctement propagé. Dans le

cas où il y a une erreur de propagation, comme à tout moment, l'utilisateur peut arrêter la propagation, soit pour ajouter de nouveaux marquages soit pour corriger un marquage qui n'a pas été correctement propagé.

- Un troisième module accomplissant la tâche d'agrégation. Il sélectionne les bons composants d'avant-plan ce qui permet de former l'objet vidéo.

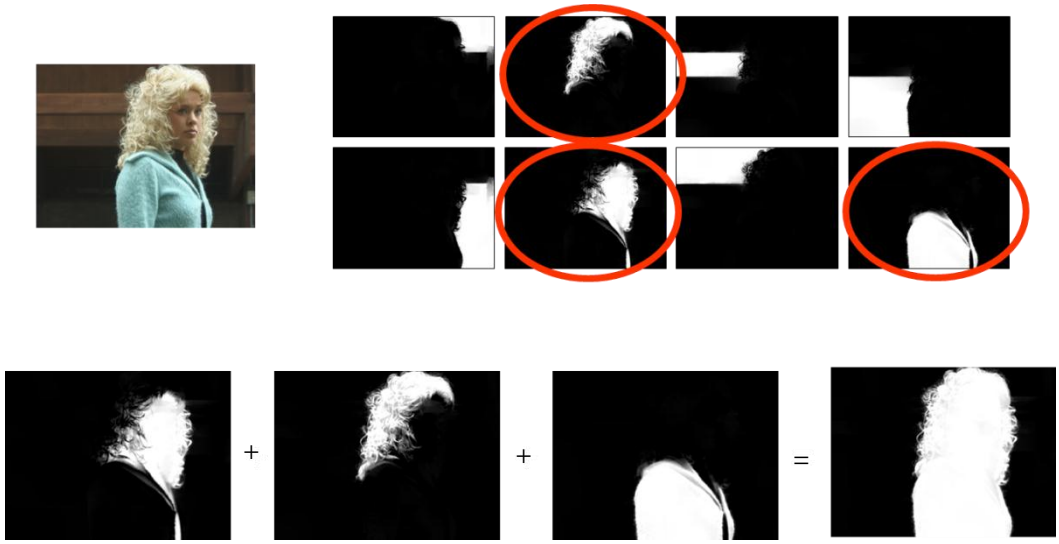


FIGURE 6 : EXEMPLE DE SUR-SEGMENTATION PAR L'APPROCHE SPECTRAL MATTING. LES SOUS-COMPOSANTS D'AVANT-PLAN PEUVENT ETRE EXTRAITS DE TOUTES LES IMAGES DE LA VIDEO, LA PROPAGATION DE MARQUAGE PERMET DE REGROUPER AUTOMATIQUEMENT LES DIFFERENTES PARTIES DE L'OBJET D'INTERET.

Cette thèse propose différentes contributions aux deux premiers modules décrits précédemment. Cela dit, nos activités se sont principalement consacrées au développement du 2ème module qui est illustré dans les chapitres 3 et 4 de ce manuscrit.

1.4 ORGANISATION DE LA THESE

Ce mémoire de thèse est divisé en cinq chapitres:

I. Dans le premier chapitre nous avons présenté le contexte et la problématique de nos travaux. Nous avons décrit les modèles interactifs et présenté notre approche et les choix faits dans la suite de nos travaux.

II. Dans le deuxième chapitre nous faisons un tour d'horizon des approches, des outils d'édition de vidéo existant sur le marché, tout en les classant en deux familles : édition par morceaux ou par blocs et édition par objets vidéo. Ensuite, nous présentons un panorama

des méthodes de segmentation, il se décompose en trois parties : la segmentation classique, la segmentation interactive et l'*image matting*. Aussi nous détaillons l'extension de l'*image matting* vers le *video matting* en présentant les principales approches existantes.

III. Le chapitre 3 présente notre première contribution pour la propagation de marquage dans les vidéos. Cette approche est une approche volumétrique 2D+t tirant sa puissance de ce que nous avons proposé une transformée en distance couleur, CDT. Nous présentons et décrivons les principes et le fonctionnement de la CDT. L'extension de la CDT à la 3D est aussi illustrée ainsi que la manière dont nous l'avons utilisée pour propager les marquages faits par l'utilisateur, dans une suite, dilatation et concentration de l'information produite.

IV. Le chapitre 4, quant à lui, présente notre évolution de perception vers un processus de propagation de marquage d'objet vidéo plus robuste et plus performant basé sur les courbes actives. Nous commençons par faire un rapide état de l'art sur les courbes actives et nous présentons par la suite notre modélisation. Elle repose sur différentes énergies mises en place pour gérer la propagation. Un mécanisme de gestion de poids dynamiques améliorant la méthode des courbes actives est aussi détaillé. Ce dernier nous a permis d'améliorer les résultats obtenus.

V. Enfin, dans le chapitre 5, nous présentons nos résultats de propagation de marquage tracé par l'utilisateur et nous verrons comment notre travail peut être inclus dans un processus rapide de *matting* vidéo ainsi que les limitations auxquelles nous avons été confronté et, aussi, quelles sont les améliorations que nous avons apportées à l'approche *Spectral Matting* pour essayer de surmonter ses limitations.

CHAPITRE 2. EDITION VIDEO

Dans ce chapitre nous faisons un tour d'horizon des approches et des outils d'édition de vidéo existant sur le marché. Nous les avons classés en deux familles : édition par morceaux ou par blocs et édition par objets vidéo. Ensuite, nous présentons un panorama des méthodes de segmentation. Il se décompose en trois parties : la segmentation classique, la segmentation interactive et l'*image matting*. Aussi nous détaillons l'extension de l'*image matting* vers le *video matting* en présentant les principales approches existantes.

2.1. **INTRODUCTION**

L'édition vidéo est une technique populairement connue sous le nom de montage vidéo. Nous pouvons préciser la signification de ce terme en nous référant à la définition qui en est donnée par le cinématographe *Serguei Mikhaîlovitch Eisenstein*[2]*:*

« Le montage est l'art d'exprimer ou de signifier par le rapport de deux plans juxtaposés, de telle sorte que cette juxtaposition fasse naître l'idée ou exprimer quelque chose qui n'est contenu dans aucun des deux plans pris séparément. L'ensemble est supérieur à la somme des parties »

Les premières formes de montage vidéo sont véritablement apparues au début du 20ème siècle, notamment avec David W. Griffith dans 'Birth of a Nation'[3] (1915). Les premières applications utilisaient un magnétoscope permettant d'enregistrer du contenu analogique provenant d'une ou plusieurs sources. Depuis l'utilisation du numérique, au cours des années 1970, les techniques d'édition et de montage vidéo ont largement évolué pour atteindre la qualification d'art ou de science selon les cas. De manière générale, si nous voulons définir le montage vidéo, nous pouvons le qualifier de l'opération qui consiste, à partir d'un ou de plusieurs contenus vidéo numériques, à créer un nouveau contenu original adapté à un scénario ou à un message particulier. Ici, nous ne parlerons pas du cas d'un montage linéaire qui était associé à l'utilisation des bandes magnétiques, mais nous allons uniquement discuter du montage numérique, aussi connu sous le nom de montage non-linéaire. Ceci nous mène plus précisément vers les problématiques associées à la manipulation et la représentation de vidéo et de ses composants. Nous pouvons citer ces quelques lignes de David Marr [Marr 1982]:

«*What does it means, to see ? The plain man's answer (and Aristotle's, too) would be, to know what is where by looking. In other words, vision is the process of discovering from images what is present in the world, and where it is. Vision is therefore an information-processing task, but we cannot think of it just as a process. For if we are capable of knowing what is where in the world, our brains must somehow be capable of representing this information -in all its profusion of color and form, beauty, motion and detail. The study of vision must therefore include not only the study of how to extract from images the various aspects of the world that are useful to us, but also an inquiry into the nature of the internal representations by which we capture this information and thus make it available as a basis for decisions about our thoughts and actions. This duality - the representation and the processing of information - lies at he heart of most information-processing tasks and will profoundly shape our investigation of the particular problems posed by vision*»

[2] http://fr.wikipedia.org/wiki/Montage
[3] http://fr.wikipedia.org/wiki/David_Wark_Griffith

Nous allons dans ce chapitre présenter brièvement l'importance de l'édition vidéo, ses origines et quelques-unes de ses spécificités. Nous allons présenter, aussi, quelques outils logiciels existant que nous classifions en deux familles : famille d'édition vidéo par blocs et famille d'édition vidéo par objets. Nous présentons les différences entre ces deux approches et nous présentons un rapide état de l'art sur la segmentation et plus spécifiquement le *matting* qui contient des techniques et des outils clé permettant aboutir à des applications d'édition vidéo réussies. Ceci met aussi en avant la nécessité d'approches telles que celles développées dans cette thèse pour simplifier la tâche aux utilisateurs et rendre les outils d'édition vidéo plus agréables et plus à la portée de l'utilisateur classique de l'outil informatique.

2.2. EDITION VIDEO : PHILOSOPHIE

Le succès et la popularité des techniques d'édition vidéo se sont développés essentiellement dans le milieu de la télévision et de la cinématographie. Cette technologie, avec ses divers outils, commence à s'installer dans les usages grand public. Plusieurs acteurs (Adobe, Apple, Microsoft…) se battent pour mettre la main sur ce marché fleurissant en offrant des outils de plus en plus simples, avec une bonne assistance, qui deviennent à la portée de monsieur tout le monde. De plus la vidéo prend une place de plus en plus grande dans notre société. Dans le domaine de l'informatique, et plus particulièrement des échanges et de la communication, la vidéo est devenue l'un des medias les plus utilisés. Lorsqu'elle est le résultat d'une acquisition, de manière générale, la vidéo peut être considérée comme issue de la capture d'un signal analogique continu. Ce signal, une fois numérisé, donne naissance à un contenu numérique appelé vidéo, support d'une information à transmettre à d'autres utilisateurs. Nous n'allons pas détailler et présenter les principes utilisés pour le codage de ce signal. Néanmoins, il est important de spécifier qu'il existe un grand nombre de formats vidéo (la famille Mpeg, Thehora, WebM, etc.). Dans le cadre d'un traitement du contenu par ordinateur, le plus souvent, ce dernier est réalisé sur les images brutes, c'est-à-dire sous la forme d'une succession d'informations statiques capturées successivement à intervalles réguliers. Les formats de codage vidéo ont un intérêt particulier pour le stockage et le transport ou la transmission. Dans le monde de l'édition vidéo, nous pouvons remarquer deux tendances majeures :

- Des approches traitant la vidéo comme un ensemble d'images qui se suivent, ce qui implique que nous pouvons éditer une nouvelle vidéo par le fait même de supprimer certaines images ou par le fait de changer leur ordre. Ces techniques sont appelées techniques d'édition de vidéo par blocs.

- D'autres approches considèrent la vidéo comme un grand nombre de pixels. Pour éditer une vidéo il faut définir des structures élémentaires qui peuvent être manipulées par l'utilisateur. Un exemple de structure peut être les points d'intérêt et donc le mouvement, [Scholz et al. 2009] cette approche a été utilisée pour ajouter des éléments s'intégrant dans la vidéo en suivant son flux d'événements. D'autres structures peuvent être considérées comme les zones homogènes, certaines textures ou simplement les objets vidéo.

2.2.1 EDITION PAR BLOCS

L'édition vidéo par blocs est la forme d'édition vidéo la plus populaire et la plus utilisée. Elle consiste à monter de courtes séquences extraites de documents variés afin de soutenir, présenter ou mettre en avant une idée, une information ou de façon plus générale un contenu. L'édition de vidéo par blocs est, en un sens, celle qui se rapproche le plus de la cinématographie traditionnelle par son système de montage linéaire. La vidéo n'est pas altérée dans son contenu mais sa personnalisation provient plutôt de la présence ou non de certaines portions et de l'ordre dans lequel les éléments de la vidéo se présentent. Bien sûr, la facilité de la combinaison des sources constitue un apport énorme permettant de transmette l'information de façon plus pertinente, en fonction du public visé et de profiter de la réutilisation de certaines parties. Selon Metz, 1974, la grande difficulté pour obtenir une vidéo de qualité réside dans les choix faits qui doivent respecter la cohérence des événements et des scènes. En effet la représentation cinématique n'est pas vraiment similaire à notre manière de rédiger ou de s'imaginer un scénario exprimé sous forme d'un texte. Le langage humain est soumis à un ensemble de règles de grammaire et respecte un certain nombre de conventions qui ne trouvent pas forcément leurs équivalents lorsqu'on passe au monde cinématographique. Il faudrait que l'utilisateur voulant produire une composition plus ou moins naturelle, prenne la peine d'assimiler un certain nombre de règles spécifiques au monde du cinéma pour obtenir un résultat correspondant à ses attentes. Il faut qu'il sache ce qu'est un *shot*, quand il faut ou il ne faut pas faire un *zoom*, gérer les *fade-in* et *fade-out*, etc., certains auteurs comme [Zancanaro et al. 2003] ont carrément essayé de modéliser un certain nombre de critères en se basant sur un ensemble de règles connues issues du monde du cinéma afin de proposer des systèmes capables de constituer des compositions de manière automatique. Les systèmes de composition de vidéo automatiques n'ont pas eu beaucoup de succès. Cela est dû à la grande variabilité des compositions possibles, à la complexité des contenus mais aussi à l'utilisateur qui a trouvé dans le web et ses nouveaux outils un monde de liberté et qui ne veut pas se créer de nouvelles contraintes entravant cette liberté qui lui est si chère. Faire un outil de composition vidéo qui soit à la

fois automatique et générique (donc qui peut au moins fonctionner pour un usage 'normal') est une tâche très complexe. Cette tâche nécessite une large palette de connaissances et, le plus complexe, un goût artistique en accord avec chaque utilisateur. Dans certains cas bien définis, la composition automatique de vidéo peut s'avérer plus performante que celle faite par une personne. En effet, elle permet de traiter rapidement de très grandes quantités de données dans des conditions bien spécifiques. Nous pouvons citer par exemple les outils utilisés pour générer des compilations des meilleurs moments d'un événement sportif. Dans [Ekin et al. 2003] les auteurs proposent un système combinant des règles basées sur une combinaison de caractéristiques de bas niveau et des caractéristiques haut niveau (détection d'un but, d'un joueur dans la zone de tir, etc.). De plus, un processus de décision permet d'identifier les meilleures combinaisons en offrant différents degrés de sélection possibles.

Hormis les cas spécifiques, la composition automatique n'arrive que rarement au niveau de qualité d'une composition vidéo faite par une personne. Plusieurs professionnels ont mis à disposition sur le marché des outils offrant aux utilisateurs classiques les moyens de base pour pouvoir rapidement effectuer des compositions vidéo.

TABLEAU 1 : EXEMPLE DE LOGICIELS D'EDITION VIDEO ET LEURS DESCRIPTIONS

Outils d'édition et de composition vidéo	**Descriptions**
VirtualDub	**VirtualDub** est un outil d'édition et de traitement de vidéo proposant plusieurs fonctionnalités (tel que l'ajout d'image ou de texte dans la vidéo, la conversion en niveaux de gris, le redimensionnent de la vidéo, la correction de contraste, …) et incluant un large choix de codec tant pour la vidéo que pour l'audio.
Adobe After Effect	**Adobe After Effect** est un outil moderne de création de contenu, il permet de mixer plusieurs sources vidéo, audio, et images.
ZS4 Video Editor	**ZS4 Video Editor** est un outil d'édition vidéo et de composition qui s'adresse aux experts en leur fournissant une facilité de combinaison entre divers types de medias (images, audio, vidéos).
Cinelerra-CV	**Cinelerra-CV** est un outil avancé d'édition vidéo non-linéaire. Il permet

	l'édition et la composition de vidéos sur les plateformes Linux. Il propose plusieurs fonctionnalités telles que la conversion de couleurs, l'ajout d'annotations ou d'images, le suivi de mouvement, etc. L'outil offre plusieurs possibilités de décompression et de compression de contenus.
Kdenlive	**Kdenlive** est un outil moderne de création de contenu, il permet de mixer plusieurs sources vidéo, audio, et images.
Blender	**Blender** est une plateforme open-source très populaire offrant plusieurs possibilités de traitement et de génération de contenus de synthèse, contenus 3D. Cet outil est disponible sur la majorité des systèmes d'exploitation. Il est distribué sous la licence GNU General Public License.
Pitivi	**PiTiVi** est un open-source écrit en python possédant plusieurs fonctionnalités d'édition vidéo telles que le changement de ratio, le changement de vitesse du contenu, la suppression de frame, etc. Ce logiciel est basé sur GStreamer et GTK+.
Avidemux	**Avidemux** est un open-source gratuit permettant le traitement et la composition de vidéo.
Jahshaka	**Jahshaka** est un outil flexible permettant d'ajouter des images et des effets spéciaux tels que l'amélioration des couleurs, le chromakeying (montage avec fond bleu ou vert), etc., et cela en temps réel à la vidéo.
Keno	**Keno** permet de charger plusieurs clips vidéo facilement. Keno permet de copier, couper et coller des morceaux vidéo et audio et permet de sauvegarder la composition sous le format SMIL XML.

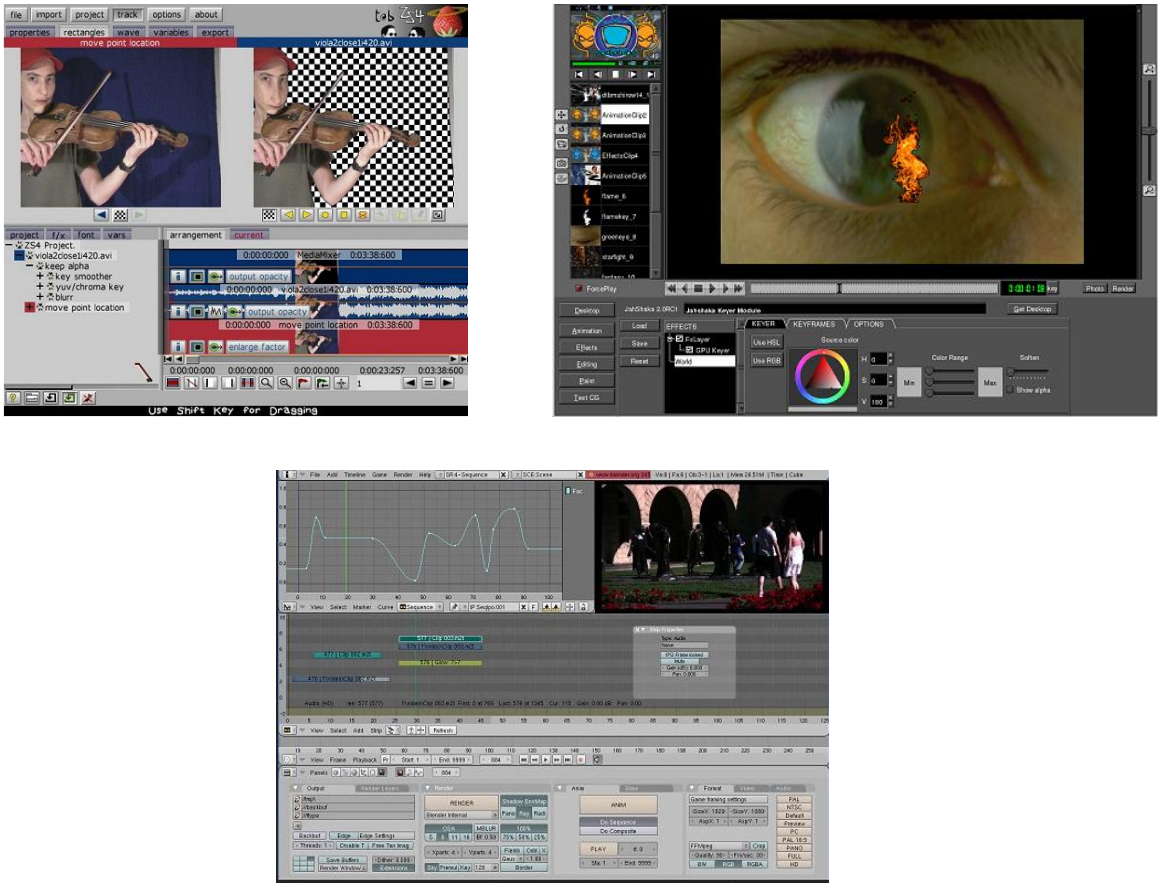


FIGURE 7 : QUELQUES CAPTURES D'ECRAN DES LOGICIELS PRESENTES DANS LE TABLEAU 1 PRESENTANT LA COMPLEXITE DE CES OUTILS

Du point de vue de la complexité d'utilisation, les logiciels présentés dans le tableau 1 peuvent être considérés comme moyennement complexes. En effet ces logiciels, pour la plupart, s'adressent à un public habitué à l'outil informatique. La Figure 7 présente un extrait des interfaces graphiques de ces logiciels. Ces interfaces confirment bien la complexité de ces outils. Le nombre de paramètres et de configurations (ce nombre dépasse la centaine pour le logiciel Bender, présenté en deuxième ligne de la Figure 7) que l'utilisation de ces outils nécessite de fixer, en atteste. Ce paramétrage demande une grande connaissance du produit et de ses détails de fonctionnement.

Pour satisfaire la demande d'un public bien plus novice et pour s'adapter aux nouvelles tendances d'applications web, d'autres outils tels que ceux listés dans le tableau 2 permettent de faire de la composition et de l'édition de vidéos dans le *cloud.* Ces outils s'adressent à un public plus novice cherchant à accomplir des tâches basiques telles que raccourcir, découper, etc. Par la délocalisation du traitement, ces outils ont l'avantage de permettre à un utilisateur quelconque de disposer d'une grande capacité de calcul (ce qui est souvent nécessaire pour le traitement de vidéo et souvent absent des

ordinateurs destinés à un usage personnel) et d'une grande capacité de stockage. De plus, ces outils permettent de nouvelles applications collaboratives entre différents utilisateurs [Outtagarts et al. 2012]. Elles permettent aussi de pouvoir faire des éditions simultanées et de pouvoir créer des contenus en partageant ses propres sources en termes de vidéos, images, etc. Les interfaces graphiques présentées dans la Figure 8 montrent la simplicité de ces applications. En effet, ces applications ne nécessitent presque pas de paramétrisation et proposent un simple système d'action : sélectionner, glisser, déposer, permettant un usage assez simpliste.

TABLEAU 2 : EXEMPLE DE LOGICIELS D'EDITION VIDEO EN LIGNE ET LEURS DESCRIPTIONS

Outils en ligne de composition de vidéo	**Descriptions**
Cuts	**Cuts** propose une interface de chargement permettant de charger ses propres vidéos ou d'en prendre sur le web comme sur YouTube, par exemple. Il propose aussi certaines animations et transitions pré-enregistrées.
Jaycut	**JayCut** est un mixeur simple et efficace permettant d'extraire et mixer des contenus vidéo et photo. Le contenu créé peut être partagé sur un réseau social tel que facebook ou myspace.
Jumpcut	**Jumpcut** est un outil d'édition et de mixage de vidéo en ligne. Jumpcut permet de créer des slideshows et de les partager sur n'importe quelle page web en permettant d'ajouter des transitions, des effets ainsi que de modifier la bande audio.
EditorOne	**EditorOne** permet de façon simple et intuitive de composer plusieurs morceaux de clips vidéo.
MovieMasher	**MovieMasher** est composé d'un ensemble d'applets Adobe Flash™ qui fournissent un outil front-end permettant d'accomplir plusieurs tâches de coupage, collage et des tâches classiques d'édition vidéo. MovieMasher propose des fonctions : trim, addition, ajout d'effet multiple sur l'audio et sur la vidéo.

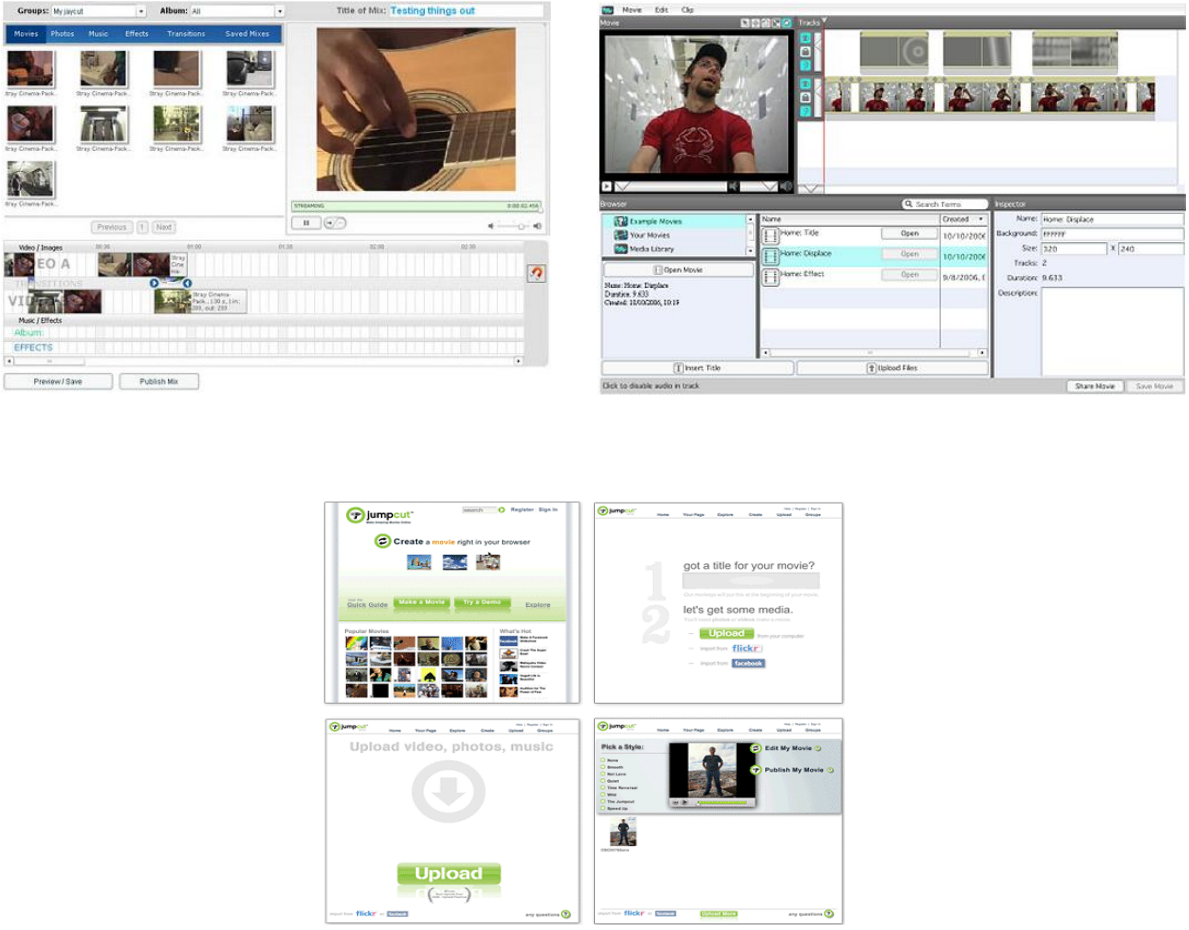


FIGURE 8 : QUELQUES CAPTURES D'ECRAN DES LOGICIELS PRESENTES DANS LE TABLEAU 2 PRESENTANT LA SIMPLICITE DE CES OUTILS

2.2.2 EDITION PAR OBJETS VIDEO

La notion d'objet vidéo est une notion assez complexe. Elle vient directement de la modélisation et de la compréhension humaine du monde naturel. La modélisation par objets a été introduite dans le monde numérique de la vidéo afin d'assurer une concordance de haut niveau entre les différents éléments issus de la représentation du monde réel et ceux contenus dans la vidéo. La vidéo représente un reflet du monde réel et non pas une vraie copie. Toutes les informations générées par le cerveau n'y sont pas présentes. Essayer de comprendre la notion d'objet vidéo, c'est essayer de reproduire la fonction que le cerveau assure, après la phase d'acquisition de l'image par l'œil, la compréhension de la scène. Néanmoins, malgré les difficultés qui se présentent, cette modélisation a beaucoup d'adeptes

qui essayent de reproduire la compréhension humaine par ordinateur. Plusieurs informations complexes pour atteindre cette compréhension sont absentes, cela découle du manque d'apprentissage, de l'absence de repères et surtout de l'absence de la 3D liée au mode d'acquisition et de restitution. Saisir et comprendre les éléments de base composant la vidéo est essentiel pour arriver à extraire des objets et ainsi identifier les éléments clé permettant d'aboutir à un processus d'édition vidéo réussi.

Transformer la vidéo en une composition, juxtaposition d'éléments, objets vidéo, nous permet d'avoir une approche plus structurelle pour l'édition de vidéo. Cette phase nécessite un traitement bas-niveau du contenu et donc une analyse au niveau de l'image elle-même.

Dans la section suivante nous allons commencer par présenter le monde de l'image numérique afin de fixer les notations et les concepts. Ceci nous permettra d'aborder plus aisément notre présentation de la segmentation d'images et du *matting*.

2.3. IMAGE NUMERIQUE

Pour bien décrire notre espace de travail, nous allons faire une brève introduction pour décrire ce qu'est la vidéo numérique. En effet une vidéo, hors du cadre de stockage et de la transmission, peut simplement être considérée plus spécifiquement comme une succession d'images couleur (dans notre cas) qui se suivent avec une certaine cadence. Après un rappel des principes de la vision humaine et ce qu'elle a impliqué pour l'acquisition des images numériques, nous aborderons des problèmes plus spécifiques, ceux de la résolution et de la représentation de la couleur ce qui introduira la notion de granularité des images et nous amènera à parler des approches de *matting* après avoir rappelé les principes de la segmentation.

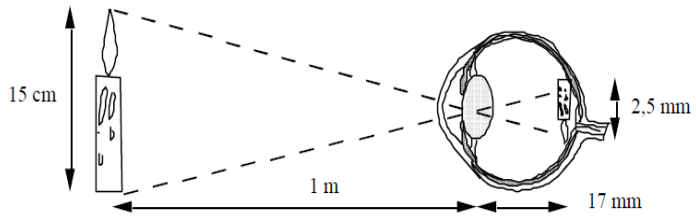
2.3.1 VISION HUMAINE

La perception humaine est, essentiellement, basée sur l'ensemble (capteur/ interpréteur) que sont l'œil et le cerveau. Le capteur, l'œil, est de conception relativement simple. Il est composé d'un ensemble de photorécepteurs appelés cônes et bâtonnets (environ 10 millions de cônes et 120 millions de bâtonnets). Les cônes sont disposés sur la paroi rétinale. Contrairement à certains capteurs électroniques, les cônes sont uniquement sensibles à une gamme spectrale limitée qui correspond à l'intervalle des longueurs d'onde de 380mm à 740mm. Les cônes peuvent être classés en trois groupes par rapport à leurs sensibilités : L, M et S. Les L-cônes sont sensibles aux longueurs d'ondes élevées,

qui correspondent aux couleurs rouges. Les M-cônes captent les longueurs d'ondes moyennes qui correspondent aux couleurs vertes. Finalement, les S-cônes sont sensibles aux plus faibles longueurs d'ondes qui correspondent aux couleurs bleues et violettes. C'est cette classification qui fait que les humains sont classés en tant que trichromatiques. Cela n'implique pas, bien sûr, que notre vision soit limitée à la perception de ces trois couleurs, mais plutôt à un sous ensemble issu des différentes combinaisons de ces dernières, ce qui forme des possibilités de presque 10 millions de couleurs distinctes. Par contre, la sensibilité de l'œil humain n'atteint pas cette précision. L'œil humain est, uniquement, capable de discerner environ 220 nuances sur une gamme de gris, rouge, vert ou bleu. Le système de perception et d'interprétation humain des couleurs est capable d'identifier la couleur d'un objet, même lorsqu'il est éclairé par des dispositifs différents.

2.3.2 VISION NUMERIQUE

La base de l'invention de la capture de photo et des appareils de photographie puise ses origines dans le système de *pin hole camera* inventé au 10ème siècle par Ibn al-Haytham, Figure 9 et 11. Une image est le résultat de l'acquisition et de la numérisation d'un signal bidimensionnel lumineux.



Distance entre cônes	:	2 à 3 µm
Acuité visuelle	:	1,5 mm à 10 m
Pouvoir séparateur	:	environ 1' d'angle

FIGURE 9 : LE PRINCIPE DE L'ŒIL HUMAIN QUI A DONNE LIEU A LA *PIN HOLE CAMERA*

La numérisation du signal image consiste à convertir les valeurs continues, analogiques, du signal en des valeurs discrètes numériques, à la fois dans le domaine spatial et dans le domaine des couleurs. Cette transformation permet d'obtenir une structure de données informatique permettant l'automatisation de certaines tâches par l'ordinateur. Elle est rendue possible grâce à des composants électroniques photosensibles qu'on appelle capteurs photographiques. Plusieurs types de capteurs permettent d'aboutir à cette conversion. Parmi les plus populaires, nous pouvons citer les capteurs CCD (*Charge-Coupled Device*, ou dispositif à transfert de charge) Figure 11, les capteurs CMOS (*Complementary metal oxide semi-conductor*) et les capteurs Foveon. Parmi ces trois familles de

capteurs, nous pouvons mettre l'accent sur les capteurs CCD, ces capteurs sont de très faible coût et assez simples de conception. Cette alliance de simplicité de fabrication avec leur bonne sensibilité a fait de ces capteurs ceux qui sont les plus répondus dans le monde. Depuis leur invention en 1969 par George E. Smith et Willard Boyle dans les Bell Labs (cette invention s'est vue accordée un prix Nobel de physique en 2009), cette famille de capteurs est maintenant solidement implantée dans les appareils photo compacts et les webcaméras.

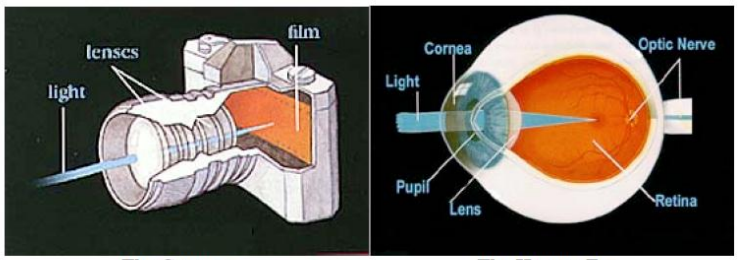


FIGURE 10 : COMPARATIF ENTRE L'ŒIL HUMAIN ET LE DISPOSITIF D'ACQUISITION D'IMAGES, LA CAMERA

Ici nous avons évoqué uniquement l'acquisition d'images et nous n'avons pas mentionné d'adaptation à la vidéo. En effet il n'y a pas eu d'avancée pour la capture et l'acquisition vidéo car la notion de vidéo elle-même repose sur la faible capacité de l'œil à voir le passage d'une image à l'autre. C'est le principe de la persistance rétinienne. L'œil humain a une persistance de 50 ms. Si les images défilent au moins à la vitesse correspondant à 12 images par seconde, cela donne l'illusion de continuité. L'absence d'information entre les images est assez courte pour que le cerveau ne remarque pas qu'il y a une discontinuité dans le mouvement des objets par exemple. Le cerveau comble l'absence de transition afin de créer une sensation visuelle de déplacement, c'est l'effet phi.

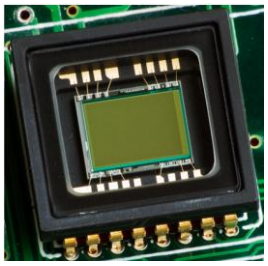


FIGURE 11 : CAPTEUR CCD

2.3.3 RESOLUTION

Une image numérique bidimensionnelle est représentée sous une forme matricielle, soit I une image à h lignes et w colonnes. Chaque élément de l'image est accessible par sa position (x, y) où, par convention, x est l'indice de largeur ($x \in [0, w-1]$) et y est l'indice précisant la position en hauteur ($y \in [0, h-1]$). L'origine du repère image est généralement fixée en haut à gauche de l'image. Chaque élément de position (x, y) est caractérisé par une valeur $I(x, y)$. Les valeurs w et h sont liées à la notion de résolution de l'image. La notion de résolution dans une image se décline en deux catégories : résolution spatiale et résolution chromatique ou dynamique de l'image. La résolution spatiale sert à mesurer la finesse et le degré de détail des éléments présents dans l'image, elle est liée à la taille réelle de ce que représente le pixel. Quant à la résolution chromatique, elle sert à indiquer le nombre de couleurs distinctes que nous pvons utiliser dans l'image. Cela impacte aussi les détails de l'image mais, généralement, impacte moins les formes (Figure 13).

A une image est associé un pavage qui peut prendre différentes formes. Généralement les pavages rectangulaires sont les plus utilisés. Ces pavages sont composés d'un ensemble d'éléments unitaires, appelés pixels (Picture Element). Nous pouvons remarquer sur la Figure 12 comment la finesse de ces éléments unitaires impacte la qualité de l'image. Si sur une certaine longueur nous avons peu de pixels, chaque pixel aura une grande dimension pour occuper l'espace et sera distinct de son voisin, l'image sera donc formée d'une série de petits carrés (pixellisation). Si sur cette même longueur nous mettons un grand nombre de pixels, ils seront d'autant plus petits qu'ils seront nombreux donc très liés les uns par rapport aux autres, et l'œil ne verra pas la limite entre deux pixels. La résolution est donc l'expression de la quantité de pixels par unité de longueur/largeur.

Résolution = nombre de pixels / pouce

640x400 pixels 150x100 pixels 38x25 pixels 10x7 pixels

FIGURE 12 : ECHANTILLONAGE, RESOLUTION SPATIALE

La dynamique de l'image est aussi relative à la notion de pixel. La couleur d'un pixel est définie par le nombre minimum de bits sur lequel ce dernier est codé. Ce nombre est donc une puissance de deux. Nous pouvons dire, par exemple, qu'une image codée en 8 bits comme celle de la Figure 13, elle contient 256 niveaux de nuances distinguables, c'est à dire légèrement plus que ce que l'œil humain peut distinguer. De manière générale, un codage sur 8 bits correspond à l'encodage d'une image en niveaux de gris et cette valeur est considérée comme la profondeur d'un canal de couleur donnée. Pour une image couleur, caractérisée en trois dimensions, la profondeur sur laquelle un pixel est codé est alors de 24 bits.

FIGURE 13 : QUANTIFICATION, RESOLUTION CHROMATIQUE

2.3.4 IMAGE COULEUR

L'analyse des images numériques couleurs couvre un champ d'investigation plus vaste que celui couvert par l'analyse des images en niveaux de gris. L'incorporation d'informations chromatiques permet aux modèles et aux modélisateurs de mieux respecter les propriétés physiques et physiologiques en concordance avec notre propre perception du monde et des couleurs. La reconnaissance d'objets sous différents éclairages constitue un exemple significatif de cette problématique spécifique à l'analyse des images couleur, un objet en couleur reflète plus d'informations et de connaissances que s'il était en niveaux de gris (Figure 14). La forme reste inchangée, les intensités sont presque les mêmes mais les luminosités changent. Cela a pour effet

qu'un algorithme différenciera deux visages au lieu de n'en reconnaître qu'un seul à cause du manque des informations chromatiques.

FIGURE 14 : EFFET DE L'ABSENCE DE LA COULEUR SUR NOTRE PERCEPTION DES OBJETS

2.3.5 AVANTAGE DU TRAITEMENT NUMERIQUE

Le traitement numérique des images est une discipline qui a révolutionné le monde. Historiquement, née aux États-Unis dans les années soixante, initialement, dans le cadre de la recherche et particulièrement avec pour objectif des applications militaires, cette discipline était assez avant-gardiste mais aussi assez déstructurée. Les chercheurs de l'époque étaient face à des problématiques nouvelles, complexes, pour lesquelles il fallait trouver les traitements adaptés. Néanmoins si les problèmes étaient bien posés, les outils techniques de calcul et d'acquisition des images n'étaient pas vraiment au point. Un exemple des problématiques de vision traitées à l'époque était de calculer automatiquement, à partir de quelques images d'une scène, la structure en trois dimensions de celle-ci. Bien sûr, ce problème est un problème mathématiquement mal posé et assez difficile puisqu'aujourd'hui encore c'est un sujet largement ouvert. Cette situation a pris fin au début des années 1980. De nouvelles théories remarquables ont fait leur apparition. Nous pouvons citer dans ce cadre le neurophysiologiste et mathématicien David Marr, chercheur au MIT (Massachussetts Institut of Technology). David Marr a proposé un modèle calculatoire pour le traitement et la représentation de l'information visuelle [Marr 1982]. Cette formalisation s'est avérée fondamentale et a donné lieu à une nouvelle manière d'aborder les problématiques de vision par ordinateur.

En plus de l'apparition de modèles précis et basés sur des abstractions mathématiques, les outils d'analyse et de traitement numérique de l'image permettent de surmonter plusieurs obstacles, que ce soit en terme d'élargissement des capacités, de nombre de traitements possibles ou aussi en terme de précision et d'exactitude de mesure (du point de vue de l'utilisation d'un repère unique).

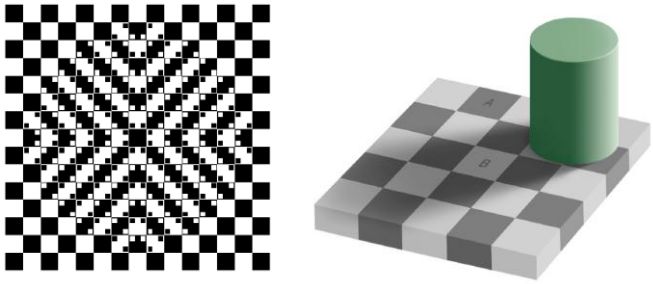


FIGURE 15 : ILLUSION D'OPTIQUE METTANT EN EVIDENCE L'IMPACT DE CERTAINES DISPOSITIONS DES ELEMENTS DANS L'IMAGE SUR L'INTERPRETATION FAITE PAR NOTRE CERVEAU

2.4. SEGMENTATION D'IMAGES

2.4.1 APPROCHES CLASSIQUES

Lorsqu'un être humain observe une scène naturelle (donc son environnement), il voit généralement des objets physiques en entier ou de façon partielle. Cela ne lui rend pas la tâche de vision plus difficile. Beaucoup d'études ont montré que l'homme était capable, jusqu'à un certain point, de discriminer les objets de manière globale même dans le cas d'occlusion, en utilisant la partie gauche de son cerveau [Vinette 2003]. Le cerveau effectue aisément la tâche qui consiste à diviser l'image en régions, objets, concepts... Cette division en elle-même peut engendrer une multitude de divisions, dans la limite du raisonnable et du visible. Nous sommes donc dans le cadre d'une interprétation dynamique qui s'adapte à plusieurs éléments tels que le contexte, les connaissances acquises, etc. Comment un ordinateur peut reproduire cette tâche en partant simplement de sa perception du monde, c'est à dire un ensemble de pixels agencés tel que nous l'avons décrit dans la section précédente ? C'est à cette question que les processus de segmentation viennent apporter une réponse.

La segmentation d'image est un processus de synthèse. C'est la tâche qui consiste à séparer l'image en différentes parties qui peuvent être qualifiées de haut niveau en partant de la richesse de l'information visuelle. Elle vise à extraire des caractéristiques géométriques des images en faisant abstraction des éléments de base, on passe du stade du pixel indépendant au stade d'une formation spécifique. Les informations contenues dans l'image peuvent être regroupées en tenant compte d'un critère de ressemblance d'intensité, de couleur ou de texture. D'autres critères d'un niveau plus élevé constituent

une problématique beaucoup plus complexe, ces critères exploitent la notion d'objet et donc sont en directe collision ou lien avec les problèmes de gap sémantique. La segmentation est une des problématiques fondamentales dans le domaine de la vision par ordinateur. Au cours de la dernière décennie, plusieurs avancées ont été mises en place et ont constitué des percées dans le domaine professionnel, industriel, médical, comme dans le domaine des loisirs et un peu aussi dans des utilisations de tous les jours. Il existe de nombreuses techniques et algorithmes de segmentation d'image. Les premières approches apparues et qui sont les plus basiques utilisent un mécanisme de seuillage d'intensité pour distinguer entre le ou les éléments d'intérêts et le fond [Sahoo et al. 1988]. Ces approches sont très populaires dans le cas de la saisie d'empreintes digitales ou de la numérisation de documents par exemple. D'autres approches un peu plus avancées utilisent la détection de contour pour déterminer la limite entre l'objet et le fond [Ziou and Tabbone 1998]. Les contours peuvent être détectés de diverses façons telles qu'en utilisant le gradient de l'image, le passage par zéro du Laplacien, etc.

La segmentation reste néanmoins un problème assez complexe, ceci est généralement dû à la grande variété des objets possibles. La variété peut être une variété de forme, de couleur, de texture, de disposition, etc. Que ce soit en utilisant des processus d'apprentissage, des heuristiques ou en se faisant aider par l'utilisateur, l'incorporation d'informations externes est un moyen efficace pour surmonter cette difficulté et permettre de concevoir des algorithmes de segmentation plus élaborés. Nous allons nous intéresser, ici, en particulier à la segmentation interactive qui a connu au cours de ces dernières années une attention particulière dans le monde de la recherche en vision par ordinateur et surtout avec l'apparition de nouvelles disciplines, en plus, telles que le '*computational photography*'. La segmentation interactive tire essentiellement son efficacité d'une connaissance assez large et non contenue dans le cadre de l'image elle-même. Ces extra-informations peuvent être fournies par l'utilisateur et ainsi simplifier cette problématique.

2.4.2 APPROCHES INTERACTIVES

Pour faciliter l'utilisation des outils de segmentation on peut imaginer qu'une solution de segmentation complètement automatique peut résoudre ce problème. La variabilité des objets, leurs morphologies ou leurs textures rendent cette tâche très difficile à maîtriser bien que dans certaines applications bien définies, c'est surtout vrai pour les applications d'imagerie médicale, il existe plusieurs méthodes [Mao et al. 2006] qui ont pu faire leurs preuves et ont pu remplacer des processus assez fastidieux.

Une solution générique mais naïve nous permettant d'accomplir des tâches de segmentation d'objet d'un niveau sémantique élevé est de mettre en place des outils permettant à l'utilisateur de détourer à la main les objets qu'il cherche à segmenter/extraire de l'image ou de la séquence vidéo considérée. De nombreuses applications médicales, ainsi que des applications de post-production télévisuelle et cinématographique utilisent cette technique. Ceci nécessite un temps énorme et un effort de maîtrise et de précision fastidieux. Une solution intermédiaire entre le tout manuel et le tout automatique s'impose. Mais la contrainte reste qu'une 'meilleure' solution doit être aussi robuste qu'une approche manuelle. Cependant elle doit offrir des temps de mise en place et d'utilisation proches des approches automatiques : cette solution est l'approche interactive.

L'objectif de la segmentation interactive est généralement de diviser l'image, de façon rigoureuse et avec un minimum d'effort de la part de l'utilisateur, en deux classes: fond et forme (cette séparation peut être parfois une séparation de type floue). Le temps d'exécution est, ici, un critère important car dans un contexte interactif, il sera attendu par l'utilisateur avant de passer à l'étape d'après car il doit valider le résultat obtenu.

2.4.3 IMAGE MATTING

La problématique de la segmentation a été pensée pour apporter des solutions à des problématiques de vision par ordinateur ce qui permet de gérer des tâches automatiquement ou de créer des systèmes robotisés. Dans le cadre des outils de traitement d'image destinés à l'édition et à la visualisation par un être humain, les résultats obtenus par les approches classiques de segmentation, visuellement, ont l'air de manquer de finition, la découpe est assez brute. Prenons par exemple la première image de la Figure 12, selon la résolution[4] que nous choisissons, l'image obtenue via des agrandissements différents, l'élément que l'on segmente va être de qualité plus ou moins bonne. Quelqu'un pourrait se dire que c'est normal car la résolution change. En réalité, ce changement de résolution, agrandissement, fait de manière numérique et non par un changement de capteur, n'apporte pas d'information supplémentaire. Simplement, les pixels sont étalés, interpolés pour prendre plus de place. Cette opération améliore légèrement l'aspect de l'élément extrait. Ceci est dû à un artefact de l'agrandissement qui est d'adoucir les frontières dans l'image. Une segmentation classique produit une découpe non naturelle sur les bords et à différentes échelles notre vision de cette netteté est différente.

[4] Par résolution nous faisons souvent référence à la résolution spatiale.

Dans le domaine de l'édition et d'extraction d'objets, durant ces dernières années, le problème de la segmentation a été reformulé d'une nouvelle façon pour répondre à de nouvelles exigences qui tournent autour de la visualisation et de l'utilisateur. Certaines étapes ou hypothèses déjà faites sont susceptibles d'être la cause du manque de finesse dans l'extraction ou le résultat obtenu. Ceci nous fait penser à la gestion des pixels lors de la phase d'acquisition suite à un certain nombre de phénomènes physiques tels que le mouvement de l'objet, le mouvement de la caméra, la réflexion de la lumière [Fox and Bertsch 2002], etc. Le problème peut même apparaître dans ce cas plus simple à cause de la difficulté que nous avons d'acquérir les informations issues du monde réel telles qu'elles sont. L'apparition d'une composante alpha, modélisant le mélange des pixels entre le fond et la forme ou de la transparence, est inévitable avec les technologies d'acquisition actuelles. Le *matting*, la problématique du tapissage, représente, donc, le processus inverse qui sert à estimer les erreurs accumulées lors de la phase d'acquisition.

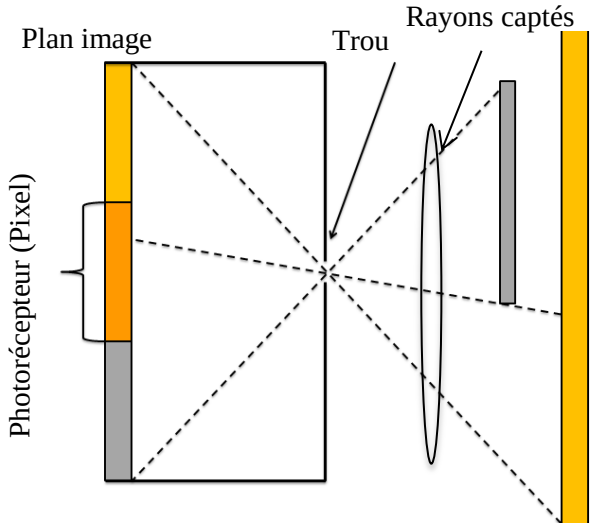


FIGURE 16 : MISE EN EVIDENCE DE L'ARTEFACT SE PRODUISANT LORS DE LA PHASE D'ACQUISITION CE QUI DONNE LIEU A UN MELANGE DE CLASSES

La segmentation classique, dans son cas binaire, peut être considérée comme un cas particulier du *matting*. En 1984 Porter et Duff ont introduit [Porter and Duff 1984], dans le problème de la segmentation, une valeur alpha permettant de contrôler linéairement la combinaison entre l'objet (à extraire) et le fond de l'image. Ainsi le problème a été défini de la façon mathématique suivante : étant donné une image I, cette image peut être décrite comme la combinaison convexe entre deux images ;

une représentant le fond B (*Background image*) et une représentant l'objet F (*Foreground image*). Cette combinaison est régulée en utilisant une valeur alpha, pour tout point de l'image on a :

$$I = \alpha F + (1 - \alpha) B \quad (1)$$

où α est une fonction à valeurs réelles dans l'intervalle [0, 1].

Donc si on veut extraire uniquement F sans tenir compte du mélange de classes, par exemple, cette opération de segmentation binaire revient alors à un cas particulier du *matting* dans lequel alpha ne peut prendre que la valeur 1.

Si on se limite à la segmentation binaire, comme son nom l'indique, le résultat qu'elle nous permet d'obtenir est composé de deux éléments constituant une partition de l'image initiale, une image fond et une image objet. Cette segmentation binaire de haut niveau semble déjà complexe. Mais grâce à l'interaction avec l'utilisateur le gap sémantique se réduit et la segmentation revient, la plupart du temps, à un problème de minimisation d'énergie. En effet, c'est l'utilisateur qui désigne l'ensemble des zones qui constituent l'objet d'étude. La résolution d'un tel problème se fait comme présenté dans [Boykov 2001] par des techniques de coupe de graphe. L'objectif du *matting* est presque le même que celui de la segmentation classique, mais le problème se pose autrement, en fait le résultat obtenu est légèrement différent, il se compose d'une image fond, là où les valeurs de alpha sont égales à 0, une image objet, là où alpha est égal à 1 et aussi une zone dite inconnue, là où les valeurs de alpha sont strictement comprises entre 0 et 1.

Le problème du *matting* est un problème mal posé. En effet, pour une image I en couleur on dispose de trois informations chromatiques (R, G, B). Du côté droit de l'équation de *matting* (1) ci-dessus le nombre d'inconnues est de 7 : trois composantes couleurs pour chacune des images B et F, et une variable alpha. Ici, nous n'allons pas présenter le cas où nous disposons d'informations externes tel que l'infrarouge, la 3D, ou des caméras élaborées, tel que celles utilisées dans [Joshi et al. 2006]. Ainsi, l'interaction avec l'utilisateur est un élément primordial pour la résolution du problème du *matting*.

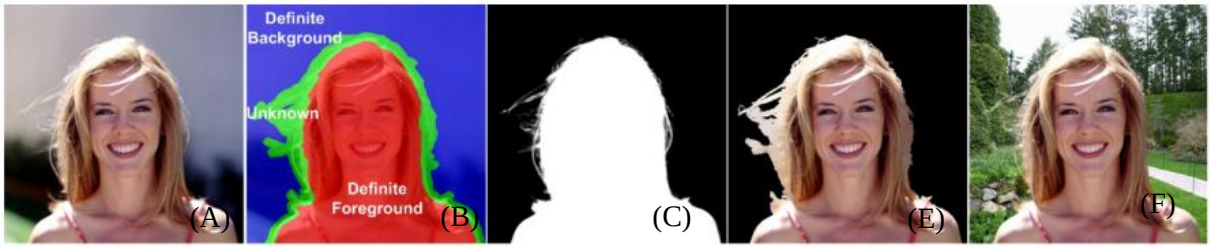


FIGURE 17 : (A) IMAGE INITIALE (B) SEGMENTATION EN TROIS ZONES '*TRIMAP*' (C) MASQUE ASSOCIE A L'ENSEMBLE DES PIXELS OU ALPHA EST STRICTEMENT POSITIF (D) (E) INCRUSTATION DE LA PERSONNE DANS UN AUTRE FOND

Supposons que nous disposons d'une pré-segmentation de l'image illustrée par la forme de Figure 17-B qui est appelée '*trimap*' [Curless et al. 2001], le problème du *matting* se restreint donc aux pixels situés dans la partie '*unknown*' (zone en vert) pour lesquels on doit déterminer la valeur de alpha. Ce type d'interaction, par *trimap*, de par son apport de point de vue d'aide aux algorithmes est le plus populaire [Curless et al. 2001; Ruzon and Tomasi 2000; Jue Wang and Michael F Cohen 2007a; Jue Wang and Michael F. Cohen 2007b; Grady et al. 2005]. La définition du *trimap* doit être assez précise, (Figure 18), plus le *trimap* est fin tout en couvrant la zone de mélange, meilleur est le résultat.

FIGURE 18 : EXEMPLE DE TRIMAP AVEC UNE DEFINITION ASSEZ PERCISECE QUI MONTRE LA COMPLEXITE D'UN TEL TYPE D'INERRACTION

Pour aussi prendre en compte l'utilisateur et lui faciliter la tâche, d'autres techniques d'interactions ont par la suite vu le jour, telle que l'interaction par gribouillis [J. Wang and M.F. Cohen 2005; Guan et al. 2006; Levin et al. 2006; Levin et al. 2008] Les gribouillis ont de plus en plus de succès (Figure 19). Ils peuvent être vus comme un *trimap* avec une région inconnue très large. Cela provoque de nouveaux challenges tels que la prise en charge de *trimap* imprécis ou la génération de *trimap* à partir de gribouillis [O. Juan and Keriven 2005].

FIGURE 19 : INTERACTION PAR *SCRIBBLES* (GRIBOUILLIS)

Il existe aussi l'interaction par rectangle (boîte englobante) [Rother et al. 2004] qui a été intégrée dans Microsoft Office 2010 (Figure 20).

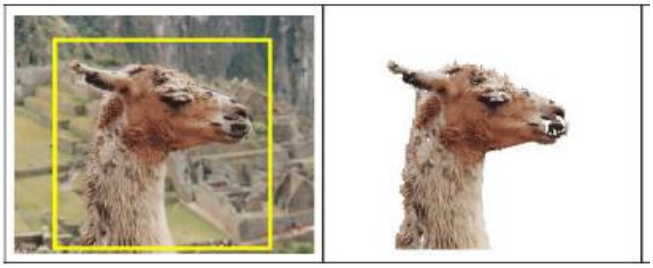


FIGURE 20 : INTERACTION PAR BOITE ENGLOBANTE

Dans la littérature, les articles et les algorithmes qui traitent la problématique du *matting* peuvent être classés en trois grandes catégories, la première catégorie contient les algorithmes basés sur l'échantillonnage de couleurs [Curless et al. 2001; Guan et al. 2006] , une deuxième utilise des descripteurs de propagation et de voisinage [Rother et al. 2004; Levin et al. 2006] et une troisième combine des descripteurs de couleur et de voisinage sous forme d'un processus d'optimisation [Jue Wang and Michael F. Cohen 2007b].

2.5 MATTING DE VIDEO

Les algorithmes de *matting* d'images ont eu une réelle attention de la part de la communauté et ils sont arrivés à un état que nous pouvons qualifier de mûr. Un grand nombre de ces algorithmes est intégré dans des solutions commercialisées. En parallèle, le *matting* vidéo a été aussi largement traité depuis un peu moins d'une dizaine d'années. La problématique de *matting* vidéo découle naturellement de

celle qui se pose dans le cas d'images fixes. Par contre, le *matting* vidéo consiste non plus à l'extraction d'objets fixes mais aussi à l'extraction des objets déformables sur des fonds qui peuvent être dynamiques. Dans le cas de la vidéo, les défis scientifiques à relever sont plus nombreux et cela est dû à la cohérence temporelle qui doit être assurée, à la quantité des données à traiter qui augmente considérablement. Pour que les algorithmes de *matting* vidéo puissent extraire des objets de haut niveau et non un ensemble de pixels partageant un critère bas niveau, ils doivent tenir compte du mouvement des objets et de l'arrière scène mais aussi du temps de traitement et du temps nécessaire pour intégrer les informations rentrées par l'utilisateur si l'on s'appuie sur une méthode interactive.

Pour pouvoir gérer toute la complexité relative au *matting* vidéo, nous pouvons diviser ce processus en deux parties : le marquage des images de la vidéo et ensuite le *matting*. Plusieurs approches demandent à l'utilisateur de marquer différentes images dans la vidéo. Ensuite les marquages sont interpolés [Bai and Guillermo Sapiro 2007; Curless et al. 2001] pour couvrir le reste de la séquence. D'autres approches introduisent le traitement de la vidéo comme un volume de données [Li et al. 2005; Jue Wang et al. 2005] pour extraire une segmentation grossière '*trimap'* afin d'y appliquer, par la suite, un processus de *matting*. Dans cette catégorie d'approches on peut remarquer l'apparition d'une nouvelle interface permettant à l'utilisateur de faire ces marquages. Cette interface, reproduite, Figure 21, n'a pas eu trop de succès car la visualisation d'un volume vidéo en 3D n'est pas assez naturelle, la reconstitution de l'image à partir de la représentation 3D demande trop d'effort à l'utilisateur.

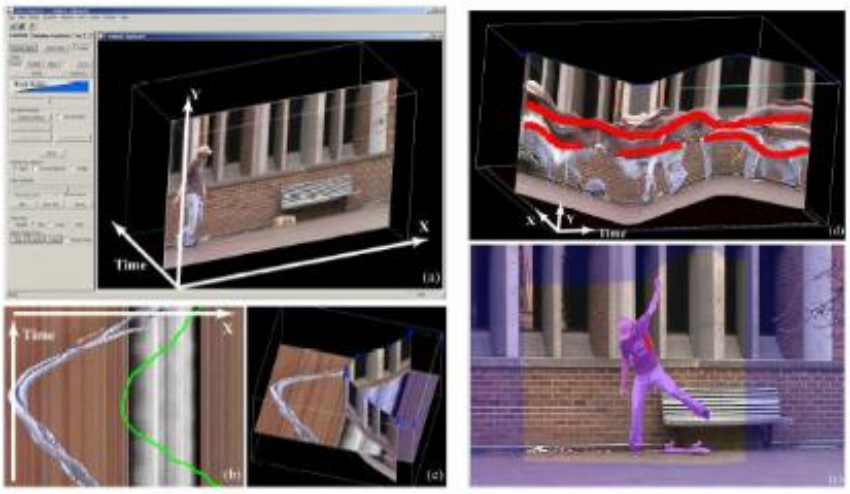


FIGURE 21 : INTERFACE D'INTERACTION ET DE VISUALISATION 3D PRESENTE DANS L'APPROCHE 'VIDEO OBJECT CUT AND PASTE'

Deux points clés ressortent de l'analyse ce processus:

- comment proposer un outil de marquage convivial et demandant le moins d'effort possible à l'utilisateur tout en fournissant un maximum d'informations utilisables?
- comment accélérer le processus de *matting* pour que le système reste utilisable sachant que, selon la méthode utilisée, la durée de l'extraction de l'objet désigné par image va d'une à plusieurs secondes ?

Le *matting* vidéo est apparu quelque temps après l'apparition des premières approches de *matting* d'images fixes. Dans [Elliott 2008] nous avons vu une extension de l'algorithme [Curless et al. 2001] par interpolation des *trimaps* rentrés par l'utilisateur en utilisant le flux optique et par reconstruction de l'arrière-plan. Parmi les premières approches prometteuses de *matting* vidéo, nous pouvons citer l'approche 'video cutout' proposé en 2005 par J. Wang [Jue Wang et al. 2005]. Cette approche représente une extension de l'algorithme de graph-cut [Boykov 2001] à appliquer directement sur un contenu vidéo. Le graphe a été construit non pas sur des images classiques mais sur un ensemble 2D+t, où la notion d'adjacence a été élaborée sur les trois axes, spatiaux et temporels. Une notion de super pixel a aussi été introduite par l'utilisation d'une pré-segmentation par *meanshift* pour accélérer le calcul. Pour l'interaction avec l'utilisateur et la définition des contraintes dures, une visualisation et une interface 3D (Figure 21) ont été proposées. Un des problèmes de cette approche réside dans le temps nécessaire à l'utilisateur pour voir un résultat, le corriger ou le valider mais le plus grand handicap venait de l'interface de marquage 3D qui offrait une visualisation très difficile et non naturelle. Dans [Bai and Guillermo Sapiro 2007] le même principe de traitement volumique est utilisé, sauf qu'ici, le volume est une sous-vidéo regroupant les frames se trouvant entre deux frames marquées par l'utilisateur. Une notion de distance géodésique a été définie par l'auteur, permettant de déterminer la frontière en partant des marquages objet et des marquages fond, cette frontière constitue le contour de l'objet. Pour avoir un traitement plus accéléré, d'autres auteurs ont utilisé le *boosting* pour apprendre les couleurs en partant des marquages indiquant l'objet et de ceux indiquant le fond. Ensuite, sans prendre en compte la notion de voisinage temporel une classification est faite sur le reste de la séquence. En 2009, X. Bai a proposé 'video snapcut' [Bai et al. 2009] qui propose une nouvelle manière d'interaction. L'utilisateur dessine la silhouette/le contour exact de l'objet d'intérêt. Par la définition d'un ensemble de cascades de classificateurs qui s'enchaînent autour de la silhouette initialement définie et l'agrégation d'un système d'estimation de mouvement (SIFT + flux optique), l'auteur a défini des modèles locaux de couleur correspondant au bord de l'objet et mis à jour au fur et à mesure de la propagation.

2.6 CONCLUSION

Dans ce chapitre, après avoir introduit le monde de l'édition vidéo ainsi que celui de l'imagerie numérique, nous avons identifié un outil clé dans ce domaine qui est la segmentation ou le *matting* d'objet vidéo. Les outils d'édition vidéo existants, sont de plus en plus populaires et c'est ce qui pousse les grands acteurs du domaine de l'édition photo et vidéo (comme Microsoft, Adobe ou Apple et d'un autre côté les universitaires) à faire la course pour fournir les briques algorithmiques et technologiques de base qui rendront cette édition un peu plus « user friendly ».

La plupart des algorithmes présentés précédemment ne sont pas assez matures pour pouvoir être intégrés dans un vrai outil commercialisable, mais ils ont contribué à faire avancer les choses de façon considérable sur ces dernières années. Cet avancement a permis d'aboutir et de voir, par exemple, Adobe intégrer la solution proposée par [Bai et al. 2009] comme un plugin à son logiciel Adobe After effect CS5. Cette solution a l'avantage de pouvoir générer un *trimap* en 1 seconde par frame, d'être robuste, et facile d'utilisation (hormis la phase d'interaction où l'utilisateur doit définir le contour exact de l'objet vidéo qu'il veut extraire).

La majorité des algorithmes existant se base sur le principe de faire intervenir l'utilisateur, en lui demandant d'intégrer des contraintes dures par l'intermédiaire d'un des outils d'interaction cités précédemment, sur plusieurs frames (ou key frame) de la vidéo et puis, soit ils appliquent un procédé d'interpolation, soit ils considèrent la sous-vidéo comme un volume vidéo et le traitent par la suite. Ils adoptent dans leur majorité deux phases :

- Une phase de génération de marquage (généralement, après une phase de segmentation binaire on génère un *trimap* par frame)
- Une phase de *matting*

Dans la suite de cette thèse nous allons nous intéresser au processus de marquage des vidéos qui constitue notre principal sujet d'étude. Nous allons avoir comment automatiser le marquage d'un objet d'intérêt en partant d'une première frame marquée par l'utilisateur. Nous allons considérer deux manières de modéliser un marquage, 'scribble', et détailler comment ces modélisations sont utilisées pour propager ces extra-informations rentrées par l'utilisateur.

CHAPITRE 3. PROPAGATION DE MARQUAGE PAR EMPREINTE VOLUMIQUE

Dans ce chapitre, nous présentons notre première modélisation de la propagation de marquages faits par l'utilisateur dans les vidéos pour la désignation d'objets d'intérêt. Cette modélisation faite est sous la forme d'une approche volumétrique 2D+T tirant sa puissance de la transformée en distance couleur (CDT) que nous avons élaborée. Nous présentons et décrivons les principes et le fonctionnement de la CDT. L'extension de la CDT à la 3D est aussi illustrée ainsi que la manière dont nous l'avons utilisée pour propager les marquages faits par l'utilisateur, dans une suite de dilatations et concentrations de l'information produite.

3.1. INTRODUCTION

L'objectif de nos travaux dans cette thèse est de propager le long des images d'une vidéo un marquage fait par l'utilisateur pour désigner un objet d'intérêt. Pour offrir un système simple est non contraignant pour un utilisateur novice, le marquage se fait de préférence uniquement sur la première frame de la séquence ou chaque fois où cela est nécessaire. Pour apporter une solution à cette problématique, nous proposons une propagation de type global sur la séquence vidéo. En ce sens, elle se fait en même temps selon les axes spatiaux et temporels de la vidéo. L'outil d'interaction que nous avons choisi, du fait qu'il semble le plus naturel et le plus simple possible pour la désignation telle que décrite précédemment, est le gribouillis. Marquer un objet par un ensemble de traits demande moins d'attention que de définir le contour de ce dernier. Ce marquage est fait sous la forme d'un dessin, une courbe qui n'a pas de caractéristiques ou de propriétés prévisibles. Dans l'optique de la facilitation de la tâche de l'utilisateur et pour réduire ses interventions, en partant du marquage initial nous proposons une modification de la transformée en distance classique qui sera désignée par le terme de transformation en distance couleur, notée CDT. Cette transformation nous permet de mettre en évidence des objets vidéo dans un volume 2D+T à partir de leur simple désignation dans la première image de la séquence en question. L'objet n'est pas connu, seule une courbe le désignant nous permet d'avoir une indication sur sa topologie et sur quelques couleurs lui appartenant. Les propriétés sélectionnées par un gribouillis sont forcément étalées aussi bien sur un voisinage spatial visible à l'utilisateur que sur un voisinage temporel, approximativement imaginé, dans le sens de l'avenir, par l'utilisateur. Pour donner une consistance à ces propriétés, on veut plonger la courbe initiale servant à designer l'objet dans une surface contenue dans le volume vidéo et plus précisément dans la trace de l'objet dans chaque image de la séquence en question. Comme dans un *level set* où la courbe de contour est obtenue par l'intersection d'une surface et d'un plan, dans chaque image, le gribouillis apparaît comme l'intersection de la surface par un plan d'équation $t=Cte$. La partie visible ne fait apparaître qu'un gribouillis. La surface est une sorte de squelette, 2D+T de l'objet désigné, d'où l'idée de la transformée en distance dans un but de rétro-ingénierie. On montre d'abord que la transformée en distance classique, même étendue en 3D est insuffisante pour résoudre notre problème. On montre dans la suite que l'extension de la carte de distance classique en intégrant les couleurs nous permet de propager les marquages tout en restant indépendant des outils d'estimation de mouvement qui sont souvent utilisés dans ce genre de propos et qui sont connus pour leur tendance aux erreurs en fonction de la nature des mouvements. Dans le cas où un ou plusieurs marquages sont utilisés pour désigner un

objet, le traitement proposé peut être appliqué en traitant les courbes indépendamment les unes des autres.

Dans ce chapitre nous commençons par une présentation de la transformation en distance et nous proposons des extensions de celle-ci pour la propagation de marquage d'objet vidéo. En d'autres termes, nous présentons une approche permettant la propagation d'une courbe le long de la séquence tout en désignant toujours le même objet.

3.2. TRANSFORMATION EN DISTANCE BASEE COULEUR

Dans cette section nous allons présenter une méthode permettant de combiner les relations spatiales et les relations chromatiques entre les pixels au travers d'une transformée en distance couleur. La relation entre les pixels de l'image est une relation forte, elle est d'autant plus forte que l'on considère les pixels d'un voisinage proche. Comparer des pixels en termes de relations spatiales est intéressant. Mais intégrer en plus des relations chromatiques est encore plus intéressant car c'est ce qui donne naissance à la compréhension de la couleur dans la perception humaine. En effet la couleur d'un pixel ou l'intensité d'un point n'a de sens que si on la compare à celle de ses voisins, tel que décrit dans la section 2.3.4.

Dans cette section, en un premier temps nous commençons par la présentation d'un mécanisme mettant en avant la notion de relations spatiales et rappelons le principe de la transformée en distance, et de la distance de chanfrein. Ensuite, nous présentons l'extension de cette transformation pour intégrer les relations chromatiques entre les pixels et donc la couleur. Après avoir mis en place cette définition, nous allons montrer l'extension de cette transformation à un volume, plus particulièrement un espace 2D+T. L'utilisation de cet outil pour la propagation des marquages faits par l'utilisateur pour désigner un objet vidéo est le fil directeur de ces différentes étapes.

3.2.1 TRANSFORMATION EN DISTANCE

La transformation en distance est un outil largement utilisé dans les domaines du traitement d'image et de la reconnaissance des formes. C'est un outil qui a vu son apparition au cours des années quatre-vingt. [Borgefors 1986] et [Ye 1988] présentent le premier algorithme efficace permettant de calculer la transformée en distance d'une image. Cette transformée est une opération applicable sur des images binaires (noir et blanc/ objet et fond), elle consiste à représenter les relations de distance entre les

pixels au travers d'une image ou simplement d'un objet par rapport au fond. La transformée en distance, notée DT, aussi appelée carte de distance est généralement représentée par une image en niveaux de gris qui est en pratique normalisée entre 0 et 255.

DÉFINITION **(Carte de distance ou DT)** *On appelle carte de distance une image en niveaux de gris qui représente les relations spatiales entre les pixels d'un objet O et l'espace dans lequel il est situé I. A chacun des pixels objet de l'image on affecte la valeur de sa distance au fond, c'est à dire au point le plus proche ou inversement, à chaque pixel du fond on affecte la valeur de sa distance au point de l'objet le plus proche. Pour chaque point p de l'image on a :*

$$DT(I) = \left(I'(p)\right)_p \text{ avec } I'(p) = \begin{vmatrix} \min\{d(p,q) \mid q \notin O\} \text{ si } p \in O \\ \min\{d(p,q) \mid q \in O\} \text{ si } p \notin O \end{vmatrix} \quad (2)$$

La transformée en distance (Figure 22) est souvent utilisée comme étape préliminaire à d'autres traitements (squelettisation [Pudney 1988], érosion...). Aussi, pour des traitements plus avancés, on trouve beaucoup d'utilisation de cette transformée dans le cadre de la reconnaissance de caractère (OCR) [Simard et al. 1992], pour le lancé de rayon dans les volumes discrets [Sramek and Kaufman 2000], la comparaison d'images binaires ou dans le contexte d'*image matching* pour lequel la comparaison des images se fait sur la transformée en distance de leurs cartes de contour binarisées (Sobel, Canny, etc) [Ziou and Tabbone 1998].

DÉFINITION **(DT d'une image relativement à un objet)** *Soit une image binaire I = F+O*

$$DT_O(I) = \left(I'(p)\right)_p \text{ avec } I'(p) = \begin{vmatrix} 0 & \text{si } p \in O \\ \min\{d(p,q) \mid q \in O\} & \text{si } p \notin O \end{vmatrix} \quad (3)$$

Lorsque nous traitons des images en niveaux de gris la notion d'objet n'a plus le même sens car on se trouve confronté au gap sémantique et qu'un objet peut avoir plusieurs représentations. Pour calculer une transformée en distance sur ce type d'images, nous faisons appel à un extracteur de contour. D'où la définition suivante :

DÉFINITION **(DT d'une image couleur)** *Soit une image couleur I et CT un opérateur de détection de contours.* $CT(I)$ *est une image binaire de mêmes dimensions que I qui contient un contour comme objet, on note C(I) le contour extrait de I :*

$$DT(I) = DT_{C(I)} \circ CT(I) \quad (4)$$

On peut distinguer trois familles d'approches permettant de calculer la transformée en distance d'une image, elles s'appuient sur des approximations de la distance euclidienne :

- Approche itérative : dans cette famille la distance euclidienne est approximée de façon itérative.
- Approche par propagation de vecteur, notée SED [Danielsson 1980]: la distance est calculé par plusieurs passages sur l'image en propageant des vecteurs reliant les points de l'objet aux points les plus proches du fond.
- Approche en deux passes : La distance euclidienne est approximée simplement par deux passages : avant et arrière.

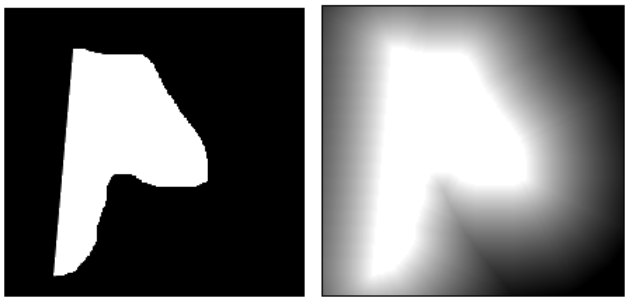


FIGURE 22 : (A) IMAGE I D'UN OBJET O (B) TRANSFORMEE EN DISTANCE $DT_O(I)$

Dans section suivante, nous allons présenter l'approche en deux passes, qui est la plus rapide. Nous détaillons le calcul des cartes de distance basé sur la distance de chanfrein.

3.2.2 CHOIX D'UNE DISTANCE

La notion de distance est généralement utilisée dans l'espace continu mais il existe aussi de nombreuses familles de distances discrètes qui peuvent être considérées dans le cadre de l'analyse d'image.

DÉFINITION **(Distance discrète)** *On appelle distance discrète sur un espace E une application :*

$d : ExE \rightarrow \mathbb{N}$ *verifiant :* $\forall\ (A, B, C)\ \in E^3$

1. $d(A,B) \geq 0$; $d(A,B) = 0 \iff A = B$

2. $d(A,B) = d(B,A)$
3. $d(A,B) \leq d(A,C) + d(C,B)$

Citons quelques distances très populaires entre deux point $A(x_a, y_a)$ et $B(x_b, y_b)$:

- Distance de Manhattan aussi connue sous le nom de city block :

$$d_4(A,B) = |x_b - x_a| + |y_b - y_a|$$

- Distance de Chessboard:

$$d_8(A,B) = \max(|x_b - x_a|, |y_b - y_a|)$$

On peut noter que la distance Euclidienne n'est pas une distance discrète.

$$d_e = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}$$

A des fins d'optimisation de calcul et pour faciliter le stockage, il est préférable d'éviter les distances à valeurs flottantes. Les distances à valeurs entières sont à privilégier. Pour plus de détails sur les distances discrètes et leurs utilisations, voir [Melter 1991].

Notons *d* une distance donnée, une approche naïve pour calculer la transformée de distance serait, pour l'ensemble des points de l'image, d'évaluer *d(p,q)* pour tout point *p* appartenant à O et tout point *q* dans *F*, et d'en déduire *d(p,F)* qui est le minimum des distances séparant *p* d'un point de *F*. La complexité d'une telle opération est proportionnelle à la cardinalité des deux ensembles : |O| × |F|. Avoir des outils algorithmiques permettant le calcul de la carte de distance de manière plus efficace est crucial.

Dans le domaine de la géométrie discrète, de nombreux auteurs ont essayé de définir des algorithmes permettant le calcul de transformations de distance pour l'approximation de la distance euclidienne. Dans la section suivante nous présenterons les principes d'une méthode d'approximation de la distance euclidienne qui est la distance de chanfrein.

3.2.3 DISTANCE DE CHANFREIN

La distance de chanfrein est une distance se calculant en propageant des distances locales par l'utilisation de masques, M, appelés masques de chanfrein. C'est une distance discrète qui consiste à

accélérer les calculs en ne considérant que des valeurs de distance entières. La distance de chanfrein minimise les erreurs en termes de moindres carrés par rapport à la distance euclidienne. Le principe de la distance de chanfrein est d'associer, en partant d'un point donné, à chaque déplacement et donc à chaque distance un coût. Ces derniers forment le masque M.

DÉFINITION **(Masque de chanfrein,** Figure 23**)** *On appelle masque de chanfrein, un masque symétrique par rapport à son centre, il contient des valeurs strictement positives traduisant le coût de chaque déplacement dans un voisinage défini par le masque.*

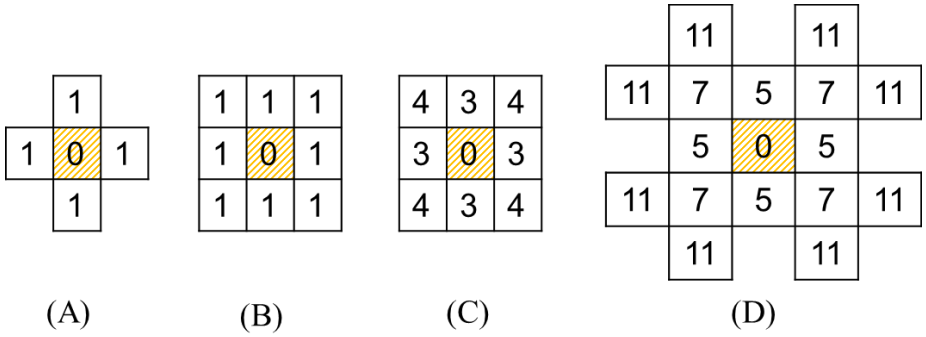


FIGURE 23 : EEMPLES DE MASQUES DE CHANFREIN

L'idée de base est d'estimer la distance euclidienne globale par la propagation des distances locales entre les pixels voisins. Pour calculer la distance entre deux points *p* et *q* dans un ensemble *E*, on considère tous les chemins possibles de *p* à *q*, formés uniquement des déplacements autorisés, c'est à dire à l'intérieur du masque qui définit les plus proches voisins du point central, et leur coût qui est la somme des coûts associés aux déplacements utilisés lors du chemin. La distance de chanfrein entre *p* et *q* est alors le minimum des coûts des chemins possibles.

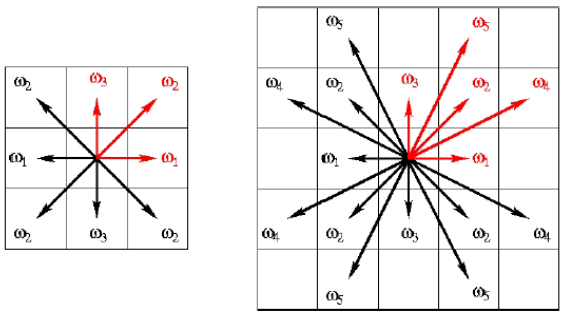


FIGURE 24 EXEMPLE DE CONSTRUCTION D'UN MASQUE DE CHANFREIN PAR SYMETRIE

Dans la pratique seule une moitié du masque de voisinage est considérée (les masques de chanfrein sont symétriques, Figure 25). Comme on le voit sur la Figure 26, un masque *M* est décomposé en deux sous-masques M_{av} et M_{ar} qui respectent l'antériorité dans un parcours d'une image ligne à ligne du haut vers le bas et sur chaque ligne de gauche à droite.

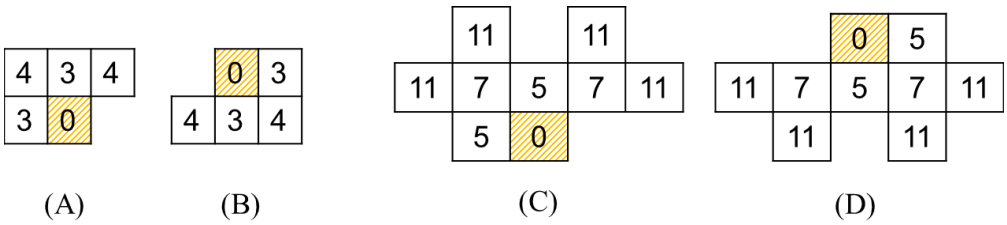


FIGURE 25 : (A),(B) DEMI-MASQUES AVANT ET ARRIERE DE TAILLE 3X3. (C), (D) DEMI-MASQUES AVANT ET ARRIERE DE TAILLE 5X5

Le calcul de la carte de distance est alors effectué en suivant un processus à deux balayages. Un balayage avant utilise la première partie du masque M_{av} et un balayage arrière utilise la deuxième partie du masque M_{ar}.

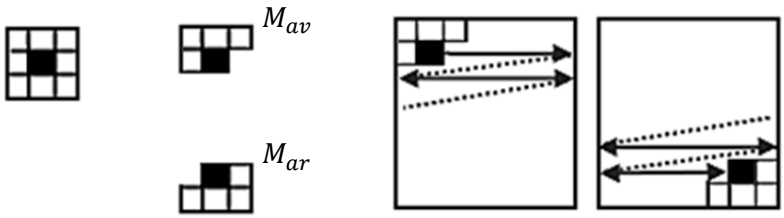


FIGURE 26 : STRATEGIE DE BALAYAGE. UNE PASSE AVANT EN UTILISANT LE DEMI-MASQUE M_{av}ET UNE PASSE ARRIERE EN UTILISANT LE DEMI MASQUE M_{ar}

L'algorithme de calclul ci-desssous présente la démarche pour le calcul de la transformée en distance en deux passes.

Entrée: Image de taille WxH plus un objet S
Résultat: Transformée en distance DT
//Initialisation
Pour j:=0 à H-1 faire
  Pour i:=0 à W-1 faire
    Si (i, j) ∈ S faire
      $DT(i, j) = +\infty$
    Sinon
      $DT(i, j) = 0$

//Passe avant
Pour j:=0 à H-1 faire
  Pour i:=0 à W-1 faire
    $DT(i, j) = min_{(x,y)\in M_{ar}}\{DT(i + x, j + y) + d_{xy}\}$

//Passe arrière
Pour j:= H-1 à 0 faire
  Pour i:=W-1 à0 faire
    $DT(i, j) = min_{(x,y)\in M_{ar}}\{DT(i - x, j - y) + d_{xy}\}$

ALGORITHME 1 : CALCUL DE LA TRANSFORMEE EN DISTANCE CLASSIQUE EN UTILISANT LA DISTANCE DE CHANFREIN

La transformée en distance classique nous permet de bâtir de la connaissance sur la forme. Pour désigner un objet nous avons besoin des informations issues de la couleur de celui-ci. Dans notre contexte les deux informations sont de la même importance : les relations spatiales et les ressemblances de couleur. La transformation en distance nous fournit une caractérisation des informations spatiales des éléments se trouvant dans l'image. Nous pouvons voir dans la section suivante comment nous avons généralisé la transformée en distance classique et défini la transformée en distance couleur pour modéliser à la fois les informations spatiales et chromatiques associées aux marquages tracés par l'utilisateur.

3.3 TRANSFORMEE EN DISTANCE COULEUR : PRINCIPE DE LA DIFFUSION 2D

Le problème que nous cherchons à résoudre n'est pas de caractériser la forme d'un objet connu mais au contraire d'identifier un objet dont nous connaissons partiellement des caractéristiques géométriques et de couleur par l'intermédiaire du gribouillis. Notre objectif est donc de créer une carte de distance qui permette de mettre en évidence des discontinuités aux limites d'un objet.

La transformation de distance classique est une transformation qui permet de bâtir une information de plus haut niveau de localisation spatiale des objets dans une image. Les informations qu'on peut extraire au travers de cette transformation nous permettent, considérant qu'une image est composée de deux ensembles, d'obtenir pour un point appartenant à un ensemble donné sa distance par rapport au premier point du deuxième ensemble. Les images en entrée de la transformée en distance sont a priori en format binaire, le seul élément pris en compte dans cette transformée, comme son nom l'indique, est la distance entre les points de chaque ensemble. L'utilisation de cette transformation sur des images couleur ne se fait qu'après un prétraitement qui associe à l'image couleur une image binaire, ce qui est le cas d'un détecteur de contours par exemple.

L'information chromatique, qui peut être contenue dans l'image, n'est pas prise en compte car la définition initiale de la DT ne le prévoit pas. Dans le cas d'images couleur, on procède par une phase de réduction de l'image à son image de contour pour pouvoir ensuite calculer la DT uniquement en utilisant les informations contenues dans l'image binaire. La partie chromatique de l'image contient des informations d'un niveau supérieur concernant les régions qui composent l'image. L'intégration de ces informations nous permettrait d'obtenir une version modifiée de la carte de distance classique qui sera plus riche en information grâce à la couleur.

Cette nouvelle carte de distance couleur (CDT) modifie les distances spatiales par l'ajout d'un terme de similarité des couleurs qui traduit par une faible distance entre deux points, leur appartenance à une même portion, car ils sont de couleur similaire, dans la scène.

DÉFINITION **(Carte de distance couleur ou CDT)** *On appelle carte de distance couleur d'une image I, une image en niveaux de gris qui représente les relations de ressemblance chromatique et spatiale dans une image I, relativement à un sous-ensemble de celle-ci, S. Elle contient, pour chacun des pixels p, la valeur d'une fonction de distance combinant les dimensions chromatique et spatiale entre p et le point de S le plus proche.*

Evidemment dans notre cas l'ensemble S représentera la courbe désignant un objet dans l'image. La même approche séquentielle en deux passes est utilisée pour obtenir une carte de distance couleur en partant d'une image de taille WxH.

Entrée: Image de taille WxH plus un marquage S, α, β
Résultat: Transformée en distance couleur CDT
//Initialisation
Pour j:=0 à H-1 faire
  Pour i:=0 à W-1 faire
    Si (i, j) ∈ S faire
      $CDT(i, j) = +\infty$
    Sinon
      $CDT(i, j) = 0$

//Passe avant
Pour j:=0 à H-1 faire
  Pour i:=0 à W-1 faire
    $CDT(i, j) = min_{(x,y)\in M_{ar}}\{CDT(i+x, j+y) + \alpha . d_{xy} + \beta . dC(p_{(i,j)}, p_{(i+x,j+y)})\}$

//Passe arrière
Pour j:= H-1 à 0 faire
  Pour i:=W-1 à0 faire
    $CDT(i, j) = min_{(x,y,)\in M_{ar}}\{CDT(i-x, j-y) + \alpha . d_{xy} + \beta . dC(p_{(i,j)}, p_{(i-x,j-y)})\}$

ALGORITHME 2 : CALCUL DE LA TRANSFORMEE EN DISTANCE COULEUR EN DEUX DIMENSIONS

Plusieurs espaces couleur et définitions de similarités couleur peuvent être considérés selon les contraintes et les problèmes traités. Nous avons, ici, utilisé l'espace RGB et la similarité couleur liée à la distance euclidienne dans cet espace à trois dimensions. Ainsi, basée sur l'algorithme de chanfrein, la transformée en distance couleur est définie pour chaque point de l'image de cordonnées *i* et *j* par:

$$\text{Initialisation : } \begin{cases} CDT\,(i,j) = 0 \quad si \quad (i,j) \notin S \\ CDT\,(i,j) = +\infty \; si \quad (i,j) \in S \end{cases} \tag{5}$$

$$CDT_I(i,j) = min_{(k,l)\in M}\left(CDT(i+k, j+l) + \alpha \cdot d\big((k,l),(i,j)\big) + \beta \cdot dC\left(p_{(i,j)}, p_{(i+k,j+l)}\right)\right)$$

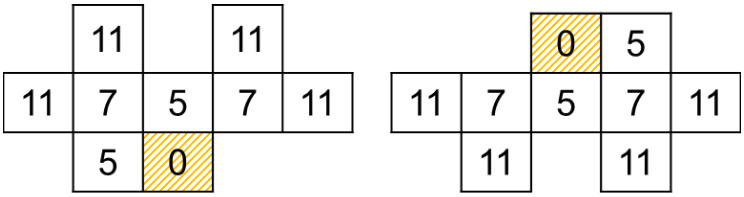


FIGURE 27 : M_{AV} M_{AR} LES DEMI-MASQUES CHOISIS POUR LA SUITE DE NOS TRAVAUX

Tel que défini dans la section 3.2.2 pour le calcul de la transformée en distance classique, le calcul de la CDT se fait en utilisant un masque de chanfrein, un exemple d'application est présenté sur les Figure 28 et Figure 29. Le masque M peut être choisi parmi les différents masques de chanfrein classiques en préservant leurs poids tel quels. Dans notre cas nous avons pris un masque 5x5 tel que décrit sur la Figure 27. Nous pouvons aussi remarquer qu'une pondération peut être faite entre la distance spatiale et la distance chromatique. Cette pondération n'a pas d'utilité dans le cas d'une carte en deux dimensions car il suffit de normaliser les deux entiers pour obtenir une carte correctement construite, nous verrons qu'il existe plus de subtilités dans le cas 3D et comment l'utilisation de ces pondération permet de combler la différence de finesse entre les résolutions spatiale et temporelle.

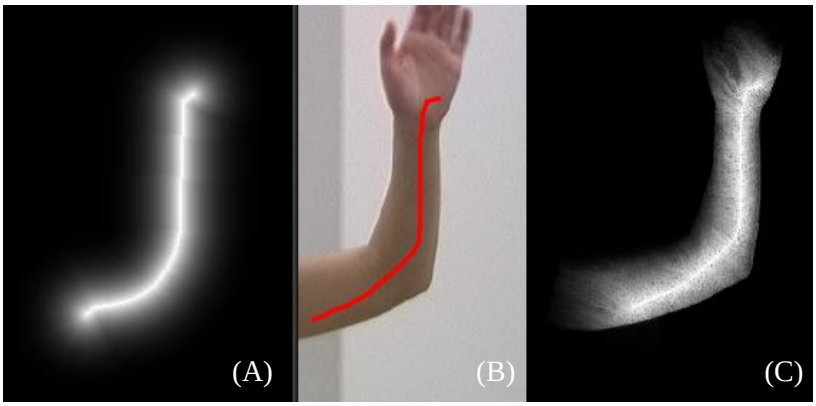


FIGURE 28 : (B) IMAGE MARQUEE, (A) DT, (C) CDT

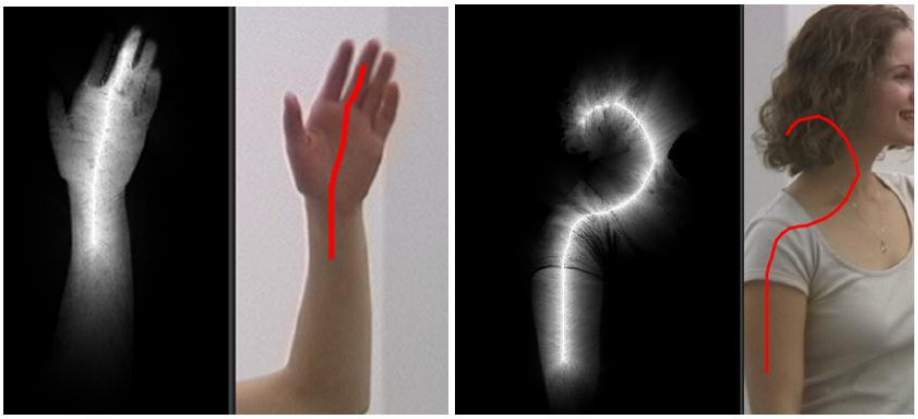


FIGURE 29 : CDT L'APPORT DE LA COULEUR POUR TRANSFORMER DES ZONES PROCHES SPATIALEMENT EN ZONES ELOIGNEES (SOMBRES) SUR LA CARTE.

Avant d'apporter une solution à la propagation intelligente de la courbe tracée sur une première image de la vidéo, il nous est nécessaire de généraliser cette transformée en distance couleur donnée dans un espace 2D et qui a un effet de diffusion d'un objet sur une surface à la diffusion dans un milieu de dimension supérieure. La généralisation concerne le volume vidéo.

3.4 GENERALISATION DE LA CDT AU VOLUME VIDEO

Après avoir précisé le principe de modélisation du marquage dans la vidéo, nous donnons la définition de la transformée en distance couleur que nous proposons dans le bloc vidéo.

3.4.1 EMPREINTE DU MARQUAGE

Quand l'utilisateur trace une courbe à l'intérieur d'un objet d'une image d'une séquence vidéo, il met son empreinte sur cet objet. On peut faire la symétrie entre cette action et l'action d'écrire sur une feuille posée au-dessus d'une pile de feuilles de papier qui contiennent chacune une copie d'un dessin visible sur la première feuille. Nous pourrions voir la trace de l'écriture apparaissant sur chaque feuille n'ayant pas pu résister à la pression. Dans le cadre de la vidéo, pour passer aux frames suivantes, s'il y avait absence de mouvement, il suffirait de matérialiser une pression 'verticale' à l'endroit de la courbe pour que la trace soit visible 'en-dessous'. La surface des traces constituerait un cylindre. Mais comme c'est plus souvent le cas, les objets d'intérêt sont en mouvement, il nous faut imaginer, alors, une force qui va modifier le sens de la résistance ou de la pression en fonction des déplacements

locaux des différentes zones de l'objet désigné. C'est principalement la couleur des points sous-jacents à la courbe qui détermine une modification de l'angle de pression.

Propager un marquage fait par l'utilisateur dans le but de désigner un objet vidéo peut être, alors, fait par l'extraction de la trace de ce marquage, de son empreinte dans le volume vidéo. L'extraction d'une trace nécessite deux informations :

- Le marquage initial.
- La carte des pressions exercées en un point et qui traduit le fait qu'un point ait une couleur identique au marquage initial.

La transformée en distance couleur va nous permettre d'optimiser la satisfaction de ces contraintes en mettant en évidence les points de plus forte pression dans les images postérieures à celle du marquage initial. La distance seule n'apporte pas d'information pertinente. Si le gribouillis est dessiné de façon proche du bord de l'objet, la propagation par distance spatiale uniquement induirait une erreur sauf si on passe par un calcul des zones de hautes fréquences, des contours, dans ce cas-là on ne dépasserait pas les bords des régions. Mais on ne pourrait pas non plus diffuser le gribouillis car l'objet qu'on veut extraire n'est pas souvent représenté par une seule zone homogène avec des bords aux alentours mais par plusieurs zones avec des critères d'homogénéité différents. Généralisée en trois dimensions, la CDT que nous allons définir, constitue un bon outil pour combiner la ressemblance chromatique et la ressemblance spatiale.

3.4.2 Principe de la CDT dans un volume 2D+T

Dans un espace 3D, la transformée en distance classique est définie de la même manière que pour les images en deux dimensions, la définition repose sur celle du voisinage considéré dans le calcul de la couleur. Son utilisation est identique, les relations spatiales sont déterminées non plus entre les pixels mais entre les voxels d'une image 3D. Dans ce contexte, la transformée en distance peut être utilisée pour déterminer le squelette d'un objet 3D. La notion de distance spatiale reste la même, on affecte un coût à chaque déplacement dans le voisinage qui est, dans ce cas, sur une grille WxHxT en trois dimensions centrée sur le point considéré. Dans le contexte d'un volume vidéo 2D+T, ce n'est pas un objet qui est composé de plusieurs couches dans le volume mais ce sont différents états de la scène qui matérialisent plutôt le déplacement d'un objet. Les deux premières dimensions, spatiales, n'ont pas la même granularité ni la même finesse que la troisième qui représente l'axe temporel de la vidéo. Le gribouillis ou le marquage fait sur la première couche doit être propagé en partant d'une information sur l'espace 2D pour atteindre le contenu du volume. C'est comme si on appliquait une pression sur un élément présentant plusieurs couches et composé de matériaux hétérogènes qui n'ont pas la même

résistance à la pression, spécialement au niveau des changements de matériau. La pression va se propager par les chemins les plus favorables, ceux le long desquels la résistance est la moins élevée. Cette résistance dans le cas de notre volume vidéo va être modélisée par une somme pondérée d'une similarité spatiale et d'une similarité chromatique que nous avons fait dépendre aussi de la profondeur de la propagation.

De manière pratique, en trois dimensions, la CDT peut être obtenue en généralisant l'approche 2D. Le masque volumique est décomposé en 2 éléments liés à l'ordre de parcours du volume. Le décrire tel que décrit ci-dessous en utilisant un masque de chanfrein 3D. Le masque utilisé est un masque 5x5x5 obtenu par symétrie centrale à partir du quart de masque présenté ci-dessous dans la Figure 30 :

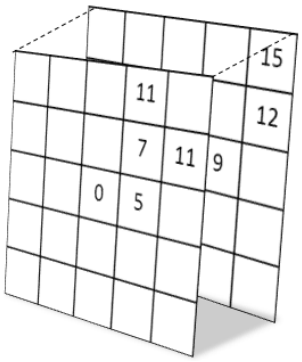


FIGURE 30 : MASQUE EN TROIS DIMENTIONS

Le marquage est contenu dans le plan caractérisé par t = 0.

$$\text{Initialisation :} \begin{cases} CDT\,(i,j,t) = 0 \quad if \quad (i,j,t) \notin marquage \\ CDT\,(i,j,t) = +\infty \; if \quad (i,j,t) \in marquage \end{cases}$$

$$CDT\,(i,j,t) = \tag{6}$$

$$min_{(k\,,l,u)\in M}\,(CDT\,(i\,+\,k,j\,+\,l,t+u) + \alpha\,.d(k\,,l)$$

$$+\beta.(u-t)\,.dC(p_{(i,j)},p_{(i\,+\,k,j\,+\,l,t+u)}))$$

L'algorithme de calcul est donné ci-dessous :

Entrée: Image de taille WxHxT plus un marquage S à t=0 , α, β
Résultat: Transformée 3D en distance couleur CDT
//Initialisation
Pour t:=0 à T-1 faire
  Pour j:=0 à H-1 faire
    Pour i:=0 à W-1 faire
      Si (i, j) ∈ S faire
        CDT(i, j) = +∞
      Sinon
        CDT(i, j) = 0

//Passe avant
Pour j:=0 à T-1 faire
  Pour j:=0 à H-1 faire
    Pour i:=0 à W-1 faire
      $CDT(i,j,t) = min_{(x,y,u)\in M_{ar}}\{CDT(i+x, j+y, t+u) + \alpha . d_{xyu} + \beta . dC(p_{(i,j)}, p_{(i+x,j+y,y+u)})\}$

//Passe arrière
Pour j:= T-1 à 0 faire
  Pour j:= H-1 à 0 faire
    Pour i:=W-1 à0 faire
      $CDT(i,j,t) = min_{(x,y,u)\in M_{ar}}\{CDT(i-x, j-y, t-u) + \alpha . d_{xyu} + \beta . (u-t) . dC(p_{(i,j)}, p_{(i-x,j-y,t-u)})\}$

ALGORITHME 3 : ALGORITHME DE CALCUL DE CDT EN TROIS DIMENSSIONS

Ci-dessous dans la Figure 31, quelques exemples nous permettent de voir que même si le marqueur (courbe) dessiné par l'utilisateur se trouve sur un objet mobile, la transformée en distance couleur, sans la nécessiter d'introduire des outils d'estimation du mouvement, permet d'avoir une idée sur le déplacement de la trace du marqueur, dans la ou les images qui suivent. A tout point p de l'image n'appartenant pas au marqueur est affectée la valeur de distance minimale le séparant du marqueur, cette valeur est d'autant plus petite que le point possède une couleur proche de celle du point du marqueur et sera grande, sinon. Ceci implique que si un point est proche spatialement mais loin chromatiquement, la valeur qui lui sera associée dans la carte de distance couleur sera grande, Figure 31. Ce raisonnement, est le même que celui de la CDT en 2D initialement présenté. Indépendamment du nombre de couches considérées, dans le cas volumique 2D+T, seul le masque est modifié par rapport à la définition 2D.

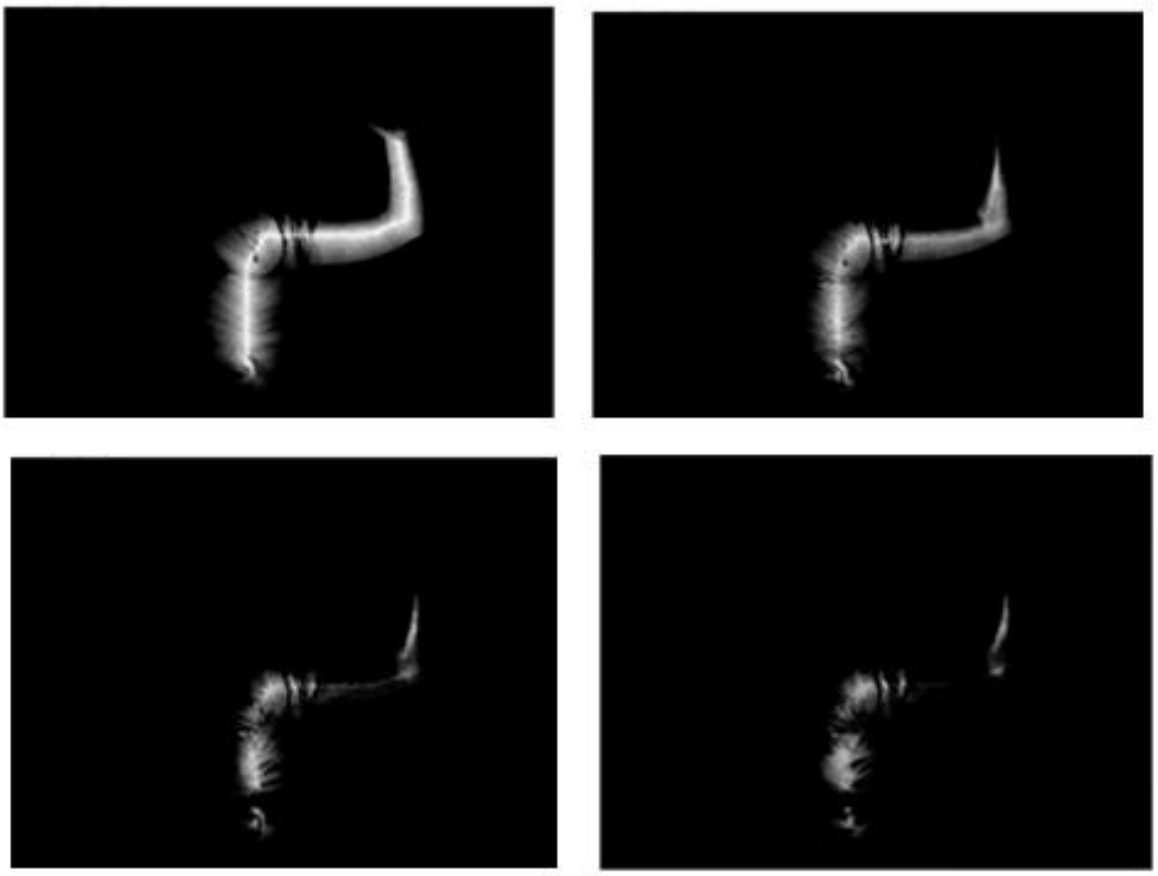


FIGURE 31 : EXEMPLE DE CDT SUR UN BLOC COMPOSE DE 4 IMAGES CONSECUTIVES

3.5 PROPAGATION D'UN MARQUAGE

Nous allons dans cette section mettre en œuvre la transformée définie dans les sections précédentes de manière à réaliser la propagation. Après avoir décrit le principe général de la méthode, nous décrirons les différentes étapes mises en évidence et nous terminerons par la mise en place d'un critère d'arrêt de la propagation nécessaire si l'objet désigné disparaît de la scène ou si une erreur de propagation se produit.

3.5.1 PRINCIPE

La propagation d'un marqueur dessiné par l'utilisateur sur la première image d'une séquence vidéo par application de la transformation distance couleur passe par deux phases :

- Diffusion de la pression du marquage et génération de son empreinte.

- Extraction de la trace du marquage sur les frames suivantes.

En effet, l'ensemble de la séquence vidéo (ou une sous-partie) peut être considérée comme une image unique en 3D (2D+T), c'est ce qu'on appelle souvent le volume vidéo.

Nous dégageons ici l'intérêt réel qu'on peut tirer de l'intégration de la chrominance dans le calcul de distance. A la différence d'une DT classique qui, pour notre volume d'images et un gribouillis donné, ne nous permet de tirer aucune information supplémentaire pour les couches en profondeur (La distance spatiale est la même quel que soit l'axe : isotropie de l'espace), la CDT va nous permettre d'acquérir une information sur la nouvelle position occupée à l'instant t+1 par l'objet désigné par le gribouillis. La propagation d'un marquage par application de la CDT sur un volume vidéo nous donne, sur les couches suivant celle où a eu l'interaction, des images correspondant à des cartes de distance couleur assez proches du bord de l'objet désigné (Figure 31). Au fur et à mesure que la propagation a lieu, les intensités des empreintes contenues dans les sections de la carte de distance couleur deviennent de plus en plus faibles et ont tendance à se dissiper. L'amplification de ces données s'avère une phase essentielle si nous voulons propager le marquage le plus loin possible. Ainsi, nous pouvons décrire le processus de propagation par des cycles de trois phases tel que décrit dans le schéma ci-dessous (Figure 32) :

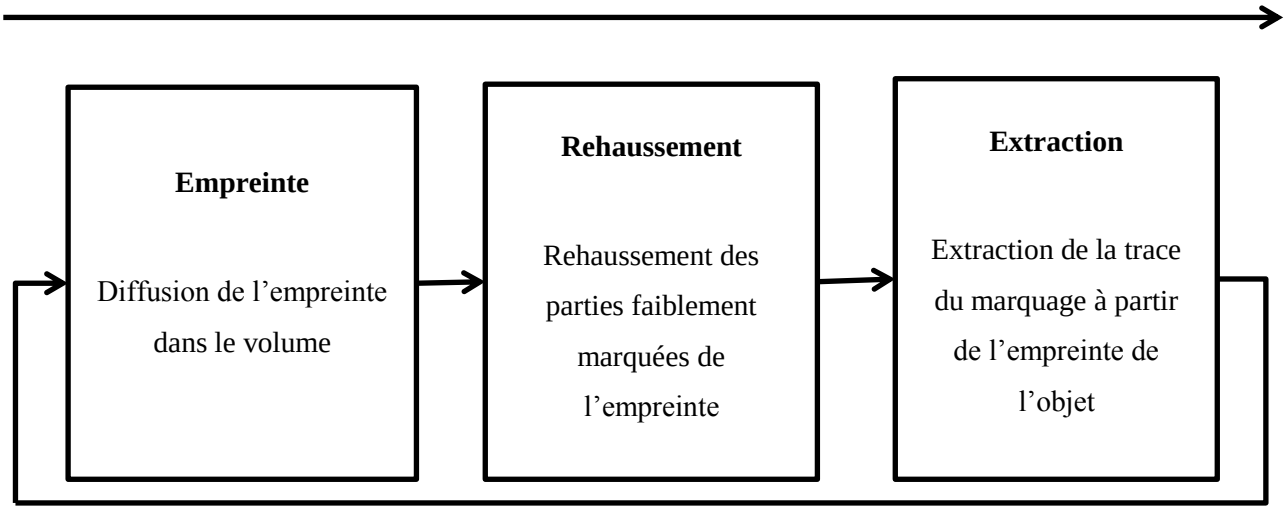


FIGURE 32 :PROCESSUS DE LA PROPAGATION ET DE L'EXTRACTION DU MARQUAGE

L'amplification des cartes de distance couleur peut se faire en appliquant la CDT successivement sur des sous-parties temporelles de la séquence d'images, donc un sous-volume vidéo (Figure 33). De chaque sous-partie on extrait un marquage final que nous utiliserons comme initialisation pour propager l'empreinte dans le sous-volume suivant. Si on considère que la propagation se fait sur un volume bicouche, cela signifie qu'on part d'un marquage sur une image *t* et qu'on travaille sur un volume V contenant deux couches, l'image I_t et l'image I_{t+1} $V=I_t \cup I_{t+1}$. Une fois la CDT calculée sur cet ensemble, on obtient un volume, noté Mcdt(V), c'est une matrice à trois dimensions de taille (WxHx2) contenant des informations sur les ressemblances de tous les pixels des deux couches aux pixels appartenant au marquage. L'idée est donc d'extraire de la deuxième couche de la matrice Mcdt(V), c'est à dire d'une carte de distance couleur, une trace d'un marquage sous la forme d'une courbe (Figure 33). On décompose donc la matrice Mcdt(V) en deux images représentant des cartes de distance couleur, $Mcdt(V)_t$ et $Mcdt(V)_{t+1}$. La première contient l'empreinte du marquage initial au temps t et la deuxième contient son empreinte au temps *t+1*. La trace est donc à extraire de la deuxième couche $Mctd(V)_{t+1}$. Cette trace sera par la suite intégrée dans le prochain sous-volume commençant au temps *t+1*. Ainsi de suite nous recommençons le même processus pour les temps *t+1* et *t+2* et pour la suite des images jusqu'à ce que le marquage soit propagé vers la fin de la séquence. Le schéma ci-dessous explicite plus concrètement le processus.

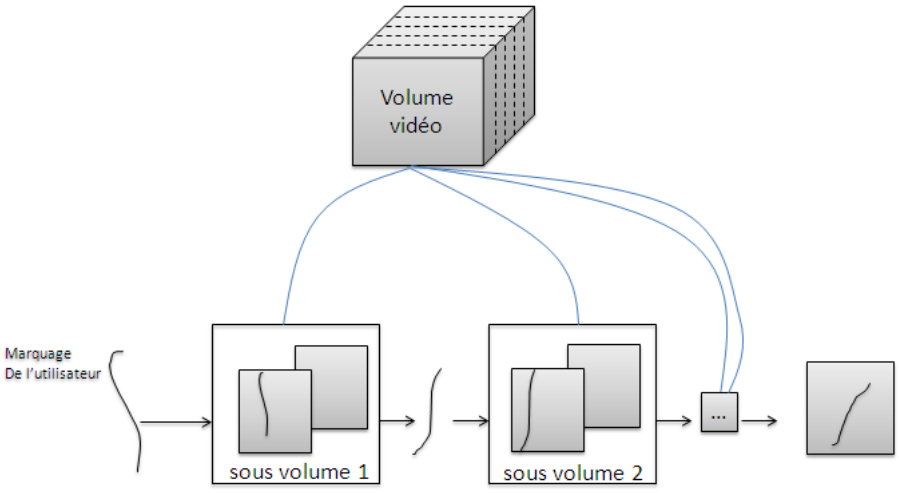


FIGURE 33 :LE TRAITEMENT DE LA VIDEO EST DIVISE EN SOUS-PARTIES POUR REDUIRE LES ERREURS D'ESTIMATION ET AVOIR DES TRACES PLUS NETTES SUR LESQUELLES TRAVAILLER

L'apparition d'irrégularités, en particulier au niveau des changements de milieu dans l'objet recherché, est fonction des données et de la distance choisie pour construire la carte des distances. Une étape de régularisation ou de rehaussement est nécessaire avant d'extraire une marque dans une image.

FIGURE 34 : COMBINAISON DE LA CDT SUR DEUX IMAGES CONSSECUTIVES

3.5.2 REHAUSSEMENT DES NIVEAUX

Une amplification par normalisation de l'énergie contenue dans chaque image ne s'avère pas efficace. Une telle approche nous permet en effet de rehausser les intensités des pixels les plus bas des couches inférieures (t>0) du volume de la CDT pour avoir des valeurs similaires aux intensités de la première couche de la CDT. Néanmoins, les voxels apparaissant comme trop loin des caractéristiques de la marque peuvent apparaître à l'intérieur de l'empreinte attendue, au niveau des transitions dans l'objet et ne sont pas comblés, Figure 34. Ces imperfections de l'empreinte que nous précisons dans la suite, nous ont amené à recourir à une étape de régularisation.

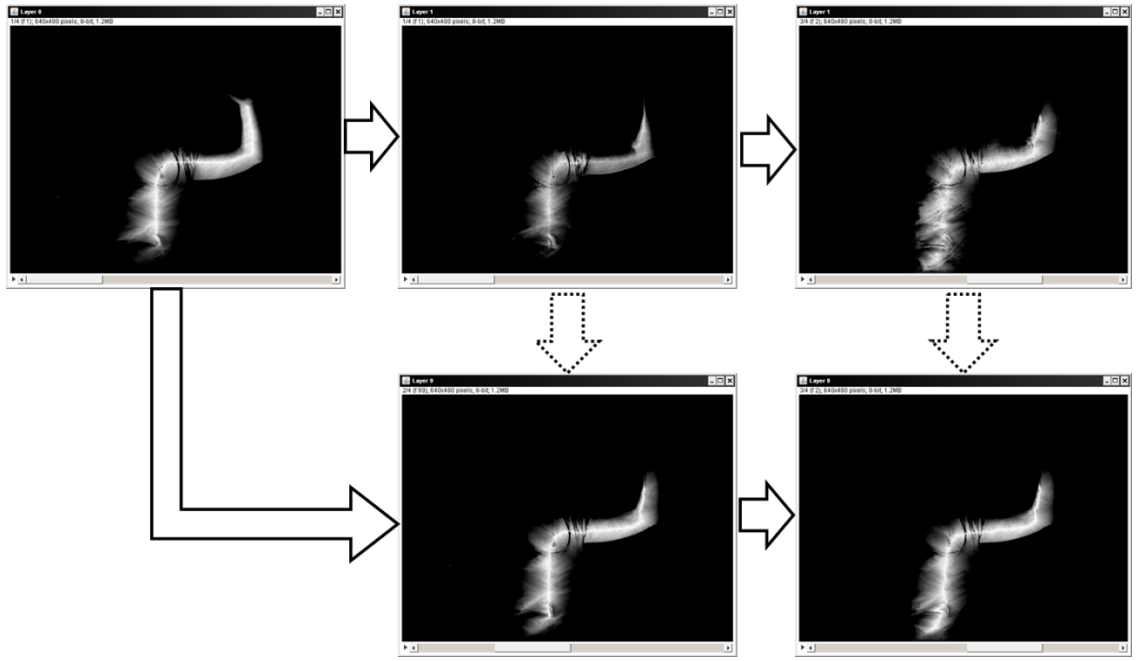


FIGURE 35 : EFFET DU REHAUSSEMENT, LA PREMIERE LIGNE PRESENTE UNE PROPAGATION SUR TROIS IMAGES SUCCESSIVES SANS REHAUSSEMENT. SUR LA DEUXIEME LIGNE ON PEUT VOIR L'EFFET DU REHAUSSEMENT SUR LES IMAGES DES COLONNES 2 ET 3.

Les images extraites du volume de pression, qui représentent les cartes de distance couleur de leurs couches correspondantes, possèdent beaucoup d'irrégularités. L'irrégularité de cette carte vient du fait que l'objet est composé de différentes régions qui peuvent avoir des mouvements différents et qui peuvent être adjacentes temporellement dans le volume vidéo avec des régions qui leur sont proches mais ne leur ressemblent pas. La carte obtenue nous permet néanmoins souvent d'inclure des pixels voisins du marquage et de se rapprocher des frontières de l'objet désigné. Utiliser une approche de binarisation classique tel que le seuillage basique des niveaux de gris ou une approche un peu plus avancée tel qu'un seuillage multi-niveaux [Quweider et al. 2007] ne permettrait pas d'obtenir de forme régulière plus adaptée pour en extraire une courbe aussi régulière que la courbe du marquage initial. De plus, si on procède par une réinjection du marquage, tel que décrit précédemment, après son extraction d'un sous-volume bicouche, l'utilisation de l'empreinte binarisée pose un problème. Sur la Figure 36, on peut voir l'impact d'une binarisation classique par seuillage, couplée avec une opération morphologique pour combler les irrégularités de la forme. Au fur et à mesure des propagations, les erreurs se propagent et s'accumulent. Elles font que l'empreinte n'est plus représentative de l'objet désigné car elle déborde, il vaut mieux avoir une empreinte fine plus proche de la forme du marquage fait par l'utilisateur pour en extraire une courbe représentative.

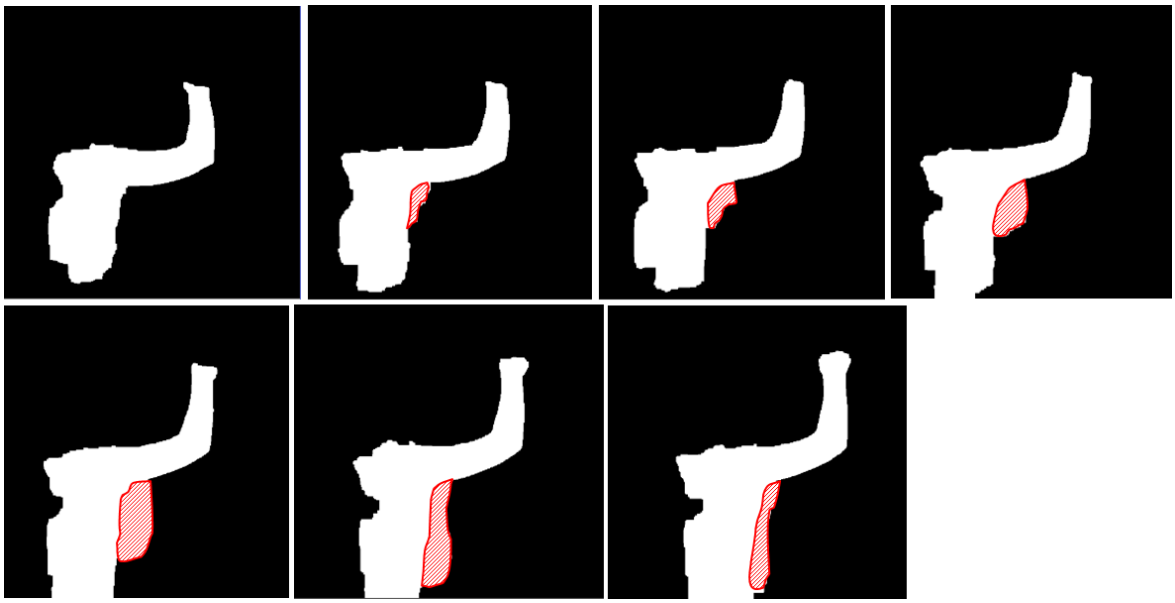


FIGURE 36 : ERREURS DE PROPAGATION DE L'EXTRACTION PAR BINARISATION

Plus qu'un rehaussement des niveaux il semble plus efficace de régulariser la forme. La carte de distance est, par construction en niveaux de gris. Un bon moyen nous permettant de combler les trous de la carte de distance couleur et aussi de capturer la topologie globale de l'empreinte présente dans la carte est d'utiliser les principes des contours actifs. L'utilisation des contours actifs est un choix qui nous permettra ici, grâce à leur possibilité d'introduire des contraintes liées à la courbe et non aux données, d'atteindre une régularisation suffisante en réintégrant les parties de l'empreinte disparues au cours de la diffusion dans les couches inférieures du volume vidéo.

Plusieurs variantes des contours actifs existent, une description plus détaillée se trouve dans la section 4.3. L'approche utilisée ici est un contour actif basé sur les champs de vecteurs gradients, noté GVF [Xu and Prince 1998], son équation discrétisée s'écrit en tout point (x, y) :

$$\begin{cases} u\nabla^2 u - (u - f_x)\left(f_x^2 + f_y^2\right) = 0 \\ v\nabla^2 v - \left(v - f_y\right)\left(f_x^2 + f_y^2\right) = 0 \end{cases} \quad (7)$$

Où ∇^2 est l'opérateur Laplacien, u et v sont deux valeurs de régularisation entre 0 et 1. f_x et f_y des carte de contour respectivement horizontale et verticale de l'image. Nous n'allons pas détailler le principe, cela dans la section 4.3, nous pouvons noter que pour notre implémentation nous avons utilisé une énergie d'uniformité assurant l'uniformité des séparations entre les points du contour, une énergie de courbure permettant au contour d'avoir des courbures assez faibles et une énergie GVF. Un critère connu pour son importance et pour l'impact qu'il a sur les résultats obtenus dans le cadre des approches par contour actif est l'initialisation de la courbe. Les images que nous avons à traiter ont des

caractéristiques assez particulières. Le fond, extérieur à l'empreinte, bénéficie d'une absence totale de bruits et, les pixels blancs indiquent l'information utile, même si ces pixels ne sont pas connectés à la zone principale. Ces contraintes impliquent que, quelle que soit l'initialisation, la courbe ne risque pas de converger vers des zones erronées. De ce fait on pourrait se permettre d'initialiser le contour de manière totalement aléatoire dans l'image. La courbe d'initialisation des contours actifs a été choisie sans intégrer de contraintes particulières, c'est le bord du rectangle maximum, le bord de l'image. D'autres approches, un peu plus adaptées, auraient pu être mises en œuvre, tel que naïvement l'utilisation de la boîte englobante de la forme, mais il est plus rapide de partir d'un contour assez large et de laisser converger le contour actif, que de calculer la boîte englobante par exemple.

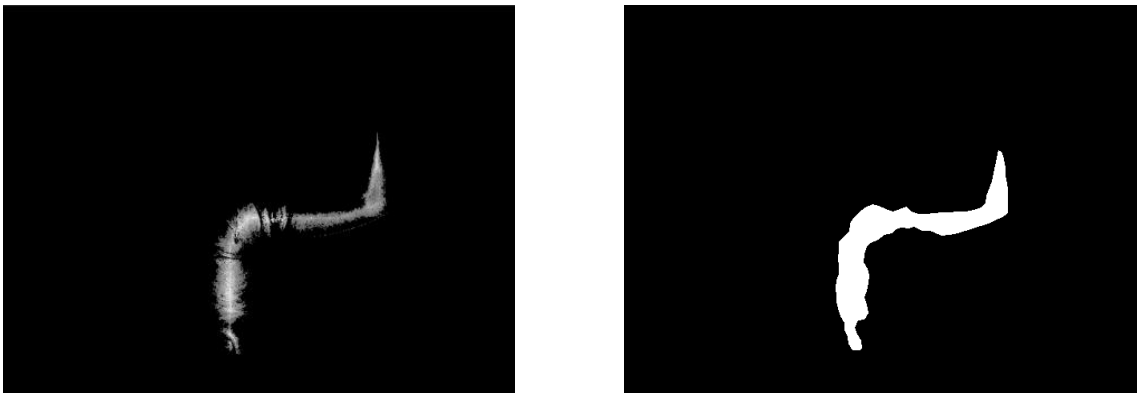


FIGURE 37 : EXTRACTION DE L'ENVELOPPE DE LA FORME PAR CONTOUR ACTIF GVF

A l'issue de cette étape, nous disposons, pour la frame à l'instant *t+1*, d'un contour fermé définissant une forme F (Figure 37), contenant une empreinte Em_{t+1} en niveaux de gris, significative de l'objet à l'intérieur de la forme F.

3.5.3 TRACE DU MARQUAGE

Pour les couches inférieures (en profondeur) du volume vidéo, une fois l'image régularisée et binarisée en fonction du contour obtenu, il s'agit d'extraire la courbe C_{t+1} la plus représentative de l'empreinte, celle qui permet de conserver un maximum d'information sur la forme F_{t+1} de l'empreinte, mais aussi

de tenir compte de la pression traduite par les niveaux de gris de l'empreinte Em_{t+1}. Nous avons choisi de procéder en deux étapes :

- Dans une première étape, nous nous intéresserons à la forme F seulement et cela en déterminant l'axe médian de l'empreinte régularisée.
- Une seconde étape permettra d'ajuster la première approximation en fonction des niveaux de gris de l'empreinte Em.

Une approche simple pour résumer une forme (binaire) en un marquage (courbe) serait d'en extraire son axe médian, son squelette, cela peut être fait à travers une des nombreuses approches de squelettisation telles que [Eberly 2001]. En utilisant une telle approche, on obtient une courbe médiane (Figure 38). Le problème de cette représentation de la forme par une telle courbe, bien qu'elle paraisse visuellement correcte en fonction de la forme de l'empreinte, est qu'elle ne coïncide pas forcément avec les extremums de la carte de distance couleur contenue dans l'empreinte. Cela signifie qu'il pourrait exister un meilleur compromis entre les deux contraintes pour que la courbe de cette image ressemble plus, dans l'esprit, à la courbe de marquage initial.

D'autres approches, tel que [Chang 2007] peuvent être utilisées pour extraire directement la trace du marquage, ou plutôt un squelette, directement d'une image en niveaux de gris or la plupart des méthodes passent par une phase de transformation en distance et de détermination de points rigides par détection de contours. Ces différents éléments font qu'il est préférable que nous élaborions une méthode propre prenant en compte les spécificités des images. On cherche en fait à extraire une courbe qui passe par les points dont les niveaux de gris sont des extremums locaux tout en tenant compte de la morphologie de la forme binaire extraite de la carte de distance couleur.

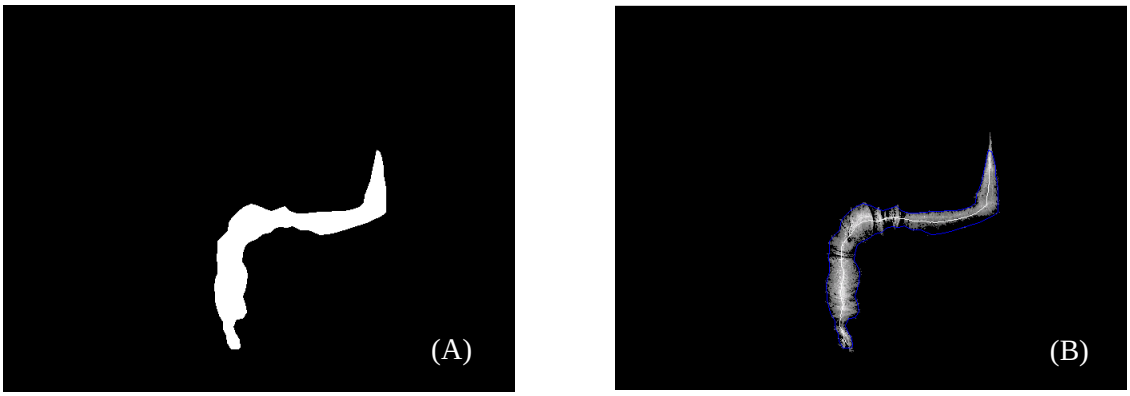


FIGURE 38 : (A) FORME F EXTRAITE PAR CONTOUR ACTIF. (B) EXTRACTION DE LA COURBE MEDIANE DE LA FORME (SQUELETTE DE F)

Comme nous le voyons sur la Figure 38, le squelette obtenu est bien situé au milieu des régions de la forme mais il ne reflète pas vraiment les extremums et donc les zones de plus forte intensité de la carte des distances couleur. Les pixels dont l'intensité est plus élevée sont les pixels les plus proches et les plus ressemblant au marquage original, tel que décrit dans la section 3.5.2. La détermination d'un axe pseudo médian s'impose alors pour faire évoluer la courbe, axe médian de la forme, vers une courbe plus représentative de l'empreinte du marquage de l'utilisateur. Cette étape peut être vue comme une optimisation de la courbe obtenue suite à la squelettisation pour satisfaire les deux contraintes précédemment posées : la représentation de la morphologie de la forme binaire et le passage par les extremums de la carte de distance couleur.

Nous considérons l'ensemble du marquage comme une suite de points. Soit $C_{squelette} = \{p_0, p_1, p_i, \dots, p_n\}$ l'ensemble des points obtenus suite à la squelettisation de la forme selon le processus décrit ci-dessus.

De manière à faire évoluer cette courbe, nous allons considérer l'ensemble de ces points comme l'initialisation d'un processus de contour actif. Au cours de ce processus nous allons considérer deux forces intrinsèques de régularisation et une force d'accroche aux données liée aux niveaux de gris de la CDT. Sur la carte de distance couleur, à chaque p_i est associée une intensité en niveau de gris. Plus cette intensité est élevée, plus on est proche des caractéristiques, en terme de similarité large, de la courbe à l'origine de l'empreinte volumétrique et donc de la carte de distance couleur. On peut ainsi définir une énergie externe représentative pour atteindre notre objectif, en un pixel *p* définie par :

$$E(p) = |\, Em(p) - V_{max}| \tag{8}$$

Où V_{max} représente l'intensité la plus élevée de la carte de distance couleur.

L'énergie d'une courbe *C* peut alors s'écrire de la manière suivante:

$$E(C) = \sum_{p_i \in C} E(p_i) \tag{9}$$

soit :

$$E(C) = \sum_{p_i \in C} |\, Em(p_i) - V_{max}| \tag{10}$$

La minimisation de cette énergie nécessite très peu d'itérations car généralement l'axe médian constitue une bonne initialisation, pour converger vers l'axe pseudo-médian passant par les extremums de la carte de distance couleur et qui est donc le plus représentatif de l'empreinte du marquage qui est à l'origine de la carte de distance obtenue sur la couche courante. Nous utilisons dans cette étape un calcul de contour actif mis en œuvre par l'algorithme glouton.

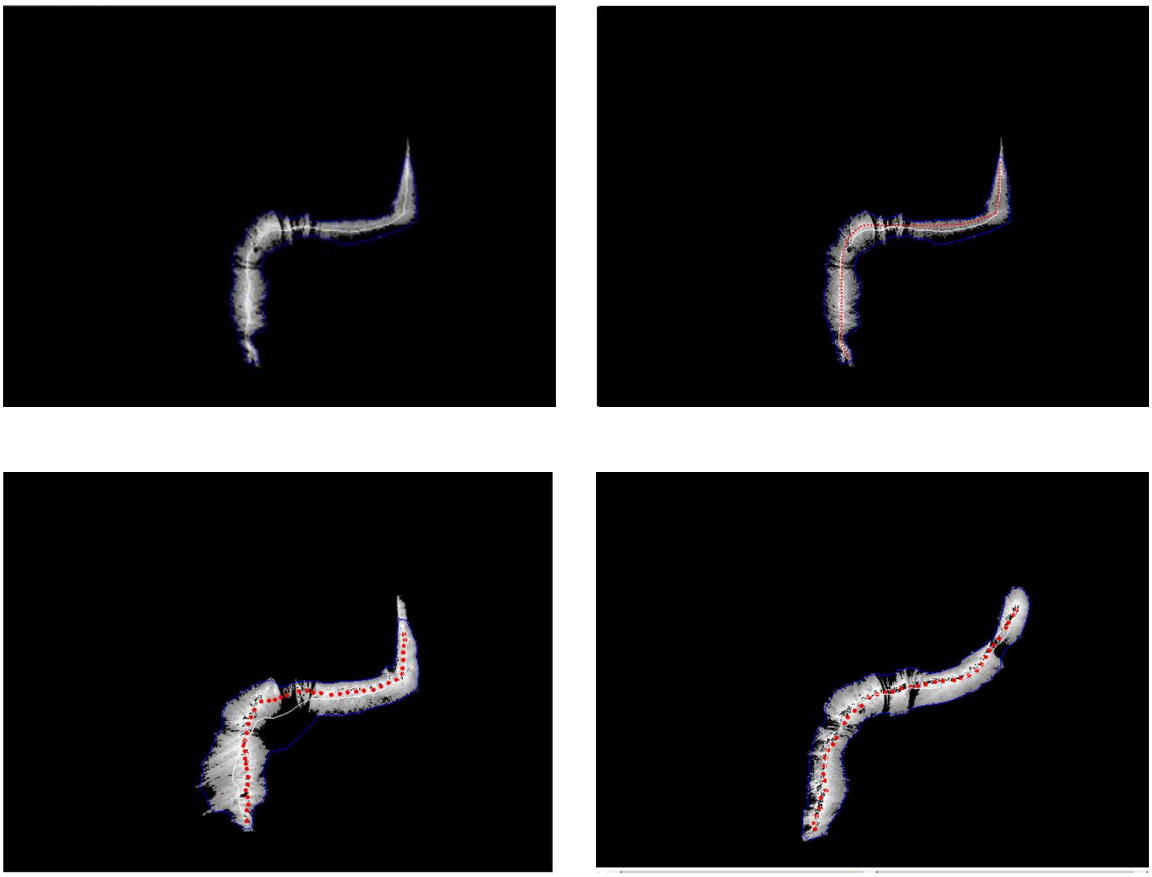


FIGURE 39 : EXTRACTION DU MARQUAGE LE PLUS REPRESENTATIF (POINTS ROUGES) EN PARTANT DU SQUELETTE (COURBE BLANCHES) DE L'ENVELOPPE DE LA FORME (CONTOUR BLEU)

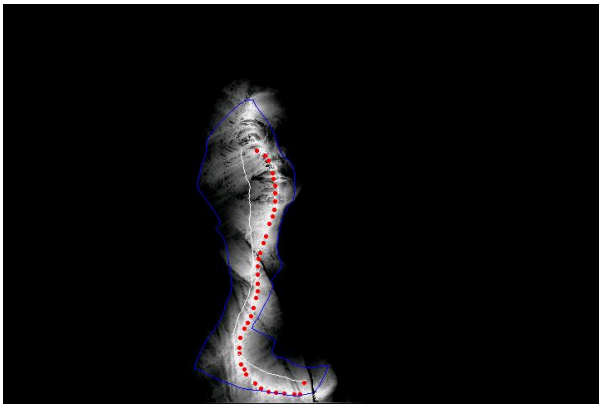


FIGURE 40 : LE SQUELETTE EST IMPACTE PAR LE BRUIT (COURBE BLANCHE). LE MARQUAGE FINAL (POINTS ROUGES) EST PLUS REPRESENTATIF DE LA CDT

Ainsi le processus décrit permet de propager une courbe sur une suite de frames. Nous noterons P_{fw} l'opérateur de propagation que nous venons de décrire. Il s'applique sur une courbe. Néanmoins, il est nécessaire de pouvoir qualifier la qualité de la propagation. La section suivante est consacrée à l'étude de qualité et à la conception d'un critère d'arrêt de la propagation quand la qualité de la trace construite n'est plus satisfaisante.

3.5.4 CRITERE D'ARRET DE LA PROPAGATION

Propager une information en partant d'une information d'origine facilite, en effet, la phase d'interaction et réduit l'effort fourni par l'utilisateur. Une mesure de la validité de la propagation peut réduire d'avantage ces efforts. Dans notre cas nous propageons un marquage. Une propagation correcte signifie que le marquage fait initialement par l'utilisateur pour désigner un objet vidéo continue à désigner le même objet. L'utilisateur peut valider si son marquage initial est correctement propagé en voyant le résultat directement sur l'écran. Il lui suffit de vérifier si le marquage est correctement positionné ou non. Cette tâche peut être faite car le cerveau possède la notion d'objet et peut facilement voir si le marquage sort de l'objet ou non, si une portion de l'objet n'est plus marquée. Du point de vue informatique, sous cette forme, cette vérification ne peut pas être automatisée car initialement nous ne connaissons pas l'objet en question. La seule information qui est disponible ici est l'image initiale dans sa globalité, la position initiale du marquage et la couleur des points qui constituent ce marquage. Disposant de ces informations, nous pouvons proposer un autre procédé pour valider la propagation.

Par application de la transformée distance couleur et en utilisant notre processus d'extraction de la trace de marquage nous pouvons déterminer vers quelle position, à l'instant *t+1*, peut évoluer un marquage S_0 dessiné en un instant *t* vers un nouveau marquage S_1. Cela peut être noté par une fonction de propagation avant notée, $P_{fw}(S_0) = S_1$.

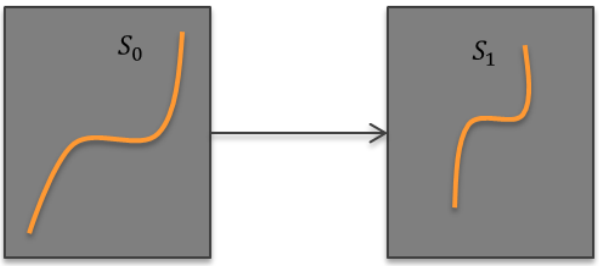


FIGURE 41 : PROPAGATION PAR P_{fw}

Pour estimer si la propagation de S_0 vers S_1 est correcte, nous avons choisi de voir à quel point notre fonction est inversible, nous avons choisi de mesurer la capacité de notre méthode à retrouver le marquage entré par l'utilisateur, S_0, à partir du marquage, S_1, obtenu par propagation. Il suffit de construire un processus P_{bw} de manière similaire à la construction de P_{fw}. Nous construisons un volume dans le sens opposé au sens de la vidéo, $v = t_{i-1} + t_i$, et nous utilisons S_1 comme entrée, marquage initial (l'équivalent du marquage fait par l'utilisateur). L'empreinte de S_1 est alors propagée par l'opérateur P_{bw}, le résultat $P_{bw}(S_1) = S_1{}'$ est une courbe déduite d'une empreinte sur la couche de la carte des distances couleur relative à l'instant t_i.

Nous pouvons alors comparer les deux marquages S_0 et $(P_{bw}{}^{\circ}P_{fw})(S_0)$ pour estimer le degré de ressemblance entre la courbe obtenue après la double propagation et le marquage tracé par l'utilisateur.

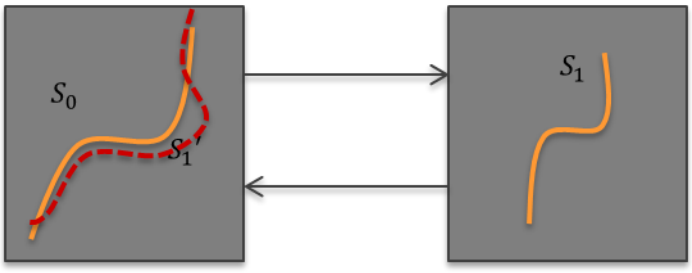


FIGURE 42 : PROPAGATION ALLER/RETOUR PAR $(P_{bw}{}^{\circ}P_{fw})$, LA COURBE RETOUR EST INDIQUEE EN POINTILLE

La trace, S_1', est extraite de l'empreinte sous la forme d'une image binaire. C'est aussi le cas pour le marquage initial. Une approche de comparaison de ces deux marquages serait de les comparer en utilisant une transformée en distance classique des deux images binaires correspondant à chacune des deux courbes. Cette approche nous permet en effet d'obtenir des informations globales sur la ressemblance des marquages. Pour être plus précis et pour pouvoir estimer la ressemblance des deux marquages de manière plus exacte, nous avons choisi de comparer les deux courbes en comparant des points se correspondant sur les courbes. S_0 étant la courbe de référence, la distance entre les deux courbes est calculée en fonction de l'écart entre les deux courbes mesuré sur la normale à la courbe en chaque point de S_0, le marquage est, suivant le cas, considéré comme une courbe ou un ensemble de points voisins, c'est en fait, une courbe discrétisée (un ensemble de points reliés entre eux en considérant un système de voisinage donné). Une fois le marquage transformé en courbe en utilisant la méthode proposée par [Damaschke 1995], par exemple, S_0 est associé à $\{pS0_0,\ pS0_1, \dots, pS0_n\}$ et S_1' associé à l'ensemble$\{pS1'_0,\ pS1'_1, \dots, pS1'_n\}$. Pour chaque point $pS1'_i$la normale à la courbe est calculée, nous la notons $NS1'_i$ (Figure 43).

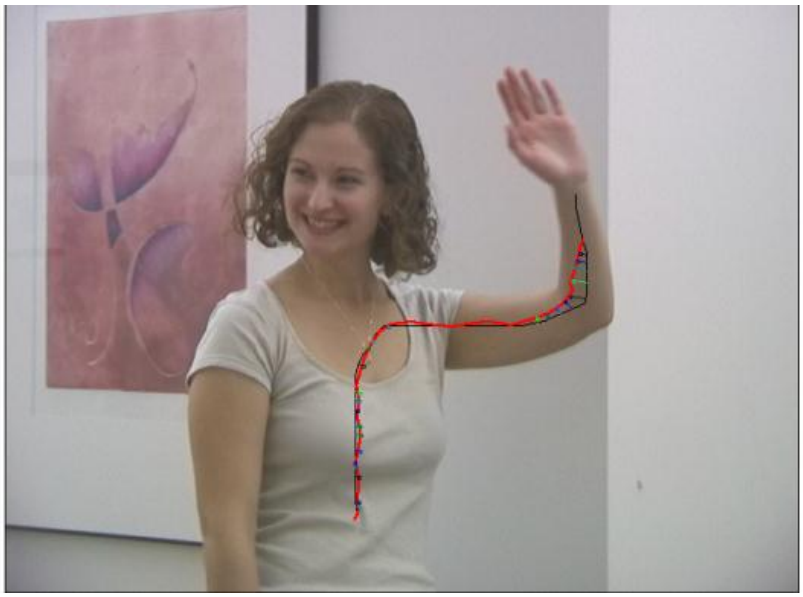


FIGURE 43 : RELATION ENTRE LES COURBES S_0 ET S_1'

La distance entre le point $pS1'_i$ et l'intersection de la courbe S_0 avec la normale à S_1' passant par $pS1'_i$ peut nous donner une information sur les erreurs de propagation. Nous pouvons considérer que la propagation du marquage se fait en deux temps, à chaque étape intermédiaire, nous considérons que le marquage que nous extrayons est un marquage correct que nous utiliserons pour la suite de la

propagation. Or, comme défini ci-dessus chaque marquage peut être propagé dans le sens inverse, ce qui permet de vérifier qu'à partir du résultat nous pouvons bien retrouver l'origine (ou au moins l'approximer). De plus, injecter ces écarts, issus de la propagation retour, dans le résultat peut améliorer la propagation. Sur la Figure 44, nous pouvons voir qu'en translatant les points de la courbe S_1 chacun par rapport aux vecteurs modélisant les écarts entre les point de S_0 et ceux de $S_1{}'$, qui la propagation retour de S_1, et S_0 (illustrer par les segments colorés reliant les deux courbes rouge et noir dans la Figure 43) :

$$M_1 \leftarrow M_1 \ + \lambda \, M_1' M_0 \tag{11}$$

Ici nous avons choisi $\lambda = 1$ mais le degré de correction ou réintégration des erreurs peut être partiel et varier entre 0 et 1. Le résultat de la propagation, dont les points corrigés sont indiqués par les points cyan, est plus représentatif (Figure 44).

FIGURE 44 : POINTS DE S1 CORRIGES EN UTILISANT LE PROCEDE D'ALLER/RETOUR

3.6 CONCLUSION

Dans ce chapitre nous avons présenté une approche de propagation de marquages en partant d'une modélisation et conception considérant un processus volumique. Ce processus est caractérisé par l'agrégation de l'information spatiale et chromatique sous la forme d'une transformée en distance couleur, CDT. Généralisé à la 3D, la CDT nous a permis de construire une connaissance de l'objet d'intérêt sur plusieurs images consécutives. Nous passons d'un marquage rentré par l'utilisateur qui peut être considéré comme un concentré de l'objet. En utilisant la CDT, nous étalons cette information

qui devient un ensemble d'empreintes imprécises. Un processus de rehaussement et d'extraction des traces à partir des empreintes obtenues nous permet d'obtenir une propagation précise d'une image à l'autre.

Les déformations du marquage initial sont obtenues successivement de manière globale. Cela peut impacter la capacité à suivre les grandes déformations issues de certains mouvements de l'objet. Pour gérer ce manque de souplesse et la difficulté que nous rencontrons dans la propagation des marquages, nous proposons dans le chapitre suivant une modélisation plus dynamique. Cette modélisation, basée sur les courbes actives, considère le marquage comme une courbe et le fait évoluer, bouger pour subir les contraintes et les données l'entourant. Nous commencerons ce chapitre par une introduction et discussion de la problématique de propagation de courbe. Nous présentons par la suite le modèle des courbes actives (connus aussi sous le nom de contours actifs). Cette présentation nous permet de rentrer plus en détails dans le sujet, de présenter notre modélisation par courbes actives et de voir son application et les améliorations que nous lui avons apportées.

CHAPITRE 4. MODELISATION DE L'ACTION DU MILIEU 2D+T SUR UNE COURBE

Dans ce chapitre, nous présentons notre évolution de perception vers un processus de propagation de marquage d'objet vidéo, plus robuste et plus performant, basé sur les courbes actives. Nous commençons par faire un rapide état de l'art sur les courbes actives et nous présentons par la suite notre modélisation. Elle repose sur différentes énergies mises en place pour gérer la propagation. Un mécanisme de gestion de poids dynamique est aussi détaillé. Ce dernier nous a permis d'améliorer les résultats obtenus.

4.1 INTRODUCTION

La propagation par transformée en distance couleur, consiste à diffuser l'empreinte du marquage fait par l'utilisateur et à extraire la trace de cette empreinte, ce qui est un processus de traitement de la vidéo sous la forme d'un volume 2D+T. Nous allons dans ce chapitre envisager une approche à plus courte portée. Dans cette approche, la propagation ne se fait plus dans un volume mais elle est réalisée progressivement d'une frame à la suivante. Nous partons d'un marquage tracé sur l'image à l'instant *t* et nous allons estimer sa position probable à l'instant *t+1*. Le marquage initial correspond à un état défini stable du marquage en considérant les données contenues dans la frame courante et un ensemble de contraintes préalablement formulées que nous allons détailler au cours de ce chapitre. L'évolution des informations ou des données dans lesquelles se trouve le marquage implique un changement de configuration. Pour revenir à un état stable, le marquage se transforme et évolue pour atteindre une nouvelle position qui traduit un nouvel état d'équilibre entre l'ensemble des contraintes et les données relatives entourant le marquage. Le passage de la frame *t* à la frame *t+1* est ainsi considéré simplement comme une évolution des données de la vidéo d'une frame à l'autre. Dans ce chapitre nous allons détailler la manière avec laquelle nous avons mis en place les mécanismes nécessaires permettant de retrouver cet état d'équilibre ce qui permet de propager le marquage initial fait par l'utilisateur.

Une modélisation possible consiste à représenter notre problématique sous la forme d'une optimisation d'un problème énergétique. Il nous faut transformer l'ensemble de contraintes issues de notre problématique que nous allons décrire dans la section 4.4 en un ensemble de forces les représentant. Il nous faut aussi, bien définir l'espace dans lequel le marquage est situé, ainsi que l'espace dans lequel ces forces évoluent. Une configuration optimale des forces, si elles sont correctement définies, permet d'atteindre, à travers une phase de minimisation, la position du marquage correspondant à une représentation optimale du marquage en partant de son état initial tout en continuant à désigner l'objet initialement marqué par l'utilisateur.

L'idée dans ce chapitre est de réduire l'impact des erreurs relatives aux estimations longues portées et de faire évoluer progressivement le marquage d'un état stable à un autre. Cela explique notre choix d'une propagation successive sur des ensembles à courte portée, d'une frame à l'autre. En effet, seules les informations provenant de la frame courante et de celle qui la suit sont utilisées pour accomplir notre processus d'optimisation.

Nous avons choisi de représenter le marquage par une courbe active. A cette courbe, différentes forces sont appliquées. Ces forces, en agissant sur la courbe, nous permettent d'obtenir, sur l'image suivante, une position satisfaisant l'ensemble des contraintes relatives à notre problème. Une énergie globale

issue de l'ensemble des forces est associée au système. La minimisation de cette énergie est le processus qui nous permet de faire évoluer notre système et ainsi faire évoluer la position de la courbe.

Dans ce chapitre, nous allons revenir sur les principes théoriques relatifs au modèle déformable des courbes actives, aussi connu sous le nom de contours actifs. Nous allons faire un tour d'horizon sur leurs différentes implémentations et les domaines dans lesquels ces approches ont été utilisées. Nous allons, par la suite, justifier et repositionner l'utilisation des courbes actives dans notre contexte de marquage d'objet vidéo ou plutôt de propagation de marquage fait par l'utilisateur. Cela nous permettra de décrire en détail notre implémentation. Nous allons décrire l'environnement dans lequel nous travaillons, en considérant les marquages comme étant des courbes, et les contraintes relatives à une telle modélisation par rapport à notre problématique. Cela nous permet d'illustrer les problèmes que nous avons résolus et de décrire la manière par laquelle nous avons modélisé des interactions de l'utilisateur par des courbes actives, comment et quelles sont les énergies que nous avons définies et associées à notre modèle. Enfin, nous décrivons comment l'évolution de la courbe a été mise en œuvre.

4.2 PROPAGATION DE MARQUAGES ET COURBES ACTIVES

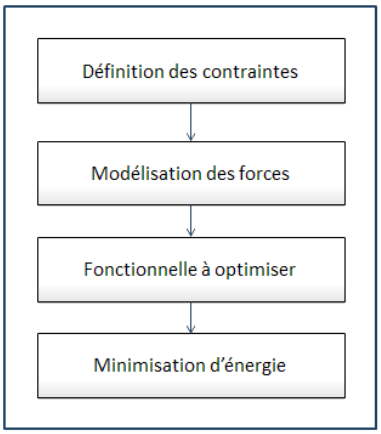


FIGURE 45 : ÉTAPES DE MODELISATION PAR COURBE ACTIVE

Comme mentionné précédemment, nos travaux sont essentiellement axés sur une étape d'extraction d'objet vidéo. Cette étape consiste en la proposition d'un système de marquage et de propagation de marquage pour la désignation d'objet vidéo. Cela par la suite, va permettre l'extraction d'objets vidéo de haut niveau et ainsi de surmonter le gap sémantique. Le système que nous proposons est un système

interactif qui permet d'intégrer les contraintes voulues par l'utilisateur tout en lui demandant le moins d'effort possible. Bien sûr nous gardons le même principe simple et intuitif d'interaction par gribouillis. Pour marquer, de manière plus simple, les objets vidéo sur l'ensemble d'une séquence, nous proposons une nouvelle méthode. En partant du marquage initialement dessiné par l'utilisateur et servant à désigner un objet d'intérêt à partir de la première image de la séquence, nous propageons d'une frame à la suivante, ces marquages, tout en permettant à l'utilisateur d'intervenir à chaque instant pour rajouter un nouveau marquage, au cas où une partie invisible de l'objet devient visible ou dans le cas où la propagation devient erronée. Cela apporte une information supplémentaire aidant à surmonter le gap sémantique. Les marqueurs permettront implicitement de définir des contraintes dures explicites pouvant être utilisées comme entrées d'un algorithme de segmentation interactive tel que [Levin et al. 2008] pour extraire l'objet final de la vidéo. Sans cette approche par propagation de l'information, un marquage sur chaque image est nécessaire pour généraliser la segmentation d'une image à l'extraction du contenu vidéo.

Nous avons réduit le problème de marquage automatique des images à un problème classique d'optimisation d'énergie. Pour détailler notre approche nous allons, en premier lieu, commencer par une présentation des contours actifs. Nous procédons par la suite à la présentation des caractéristiques et des problèmes relatifs à la propagation de courbes seront détaillés. Ensuite, nous expliciterons notre approche et nos démarches pour la résolution du problème de propagation de courbes (marquage) en tenant compte des différentes propriétés de celles-ci.

4.3 CONTOURS ACTIFS : LE PRINCIPE

Notre modélisation des marquages faits par l'utilisateur pour désigner un objet d'intérêt dans une vidéo se fait au travers d'une modélisation par contour actif où plusieurs étapes sont nécessaires avant d'aboutir à un système complet. Avant de décrire de façon rigoureuse notre modélisation et notre approche pour propager les marquages tracés par l'utilisateur, dans cette section, nous allons présenter les bases théoriques de la méthode des contours actifs. Nous ferons un tour d'horizon des différentes implémentations existantes. Les principales limitations de cette théorie sont énumérées et les solutions qui ont été proposées pour s'en affranchir sont détaillées.

Les contours actifs sont souvent connus pour leurs utilisations dans le domaine de la segmentation pour extraire un ou plusieurs objets d'intérêt dans une image (un certain type d'organe, cellules, personnes, etc.). Un autre usage des contours actifs est connu dans le cadre de la détection et du suivi

dans les vidéos (suivi de ligne blanche sur les routes [Poz et al. 2003], joueurs [Lefevre and Vincent 2004], véhicules, etc.). Ces usages se sont multipliés tant dans les traitements 2D que dans les traitements 3D. Dans suite de ce chapitre, nous allons uniquement nous intéresser à l'univers 2D qui est en concordance avec nos travaux. Le résultat obtenu par les méthodes de contours actifs n'est pas immédiat, la construction d'un contour suit un processus itératif nécessitant une initialisation préalable, une recherche d'un équilibre des énergies et une évolution du contour depuis son état initial vers une solution optimale, l'évolution est contrôlée par un test de convergence du processus. C'est à la phase d'évolution dynamique du contour, que la méthode doit sa dénomination de contour "actif". L'évolution du contour est liée à la minimisation d'une fonctionnelle d'énergie, construite de telle sorte qu'un minimum local se trouve à la frontière de l'objet à détecter.

4.3.1 PROBLEME MAL POSE

Dans le contexte où les contours actifs sont utilisés qui est celui des images numériques (ensemble fini de pixels caractérisés par une fonction d'intensité I, I(x, y) est la couleur du pixel, (x, y) varie dans un domaine borné de $\mathbb{N}^2$, typiquement lié à la résolution de l'image), l'optimisation par contours actifs est un problème mal posé. Cela veut dire qu'il existe une infinité de solutions mais elles ne sont pas forcément 'correctes'. C'est dans ce cadre initial que cette approche a été proposée initialement par [Kass et al. 1988], comme une approche de régularisation à un problème inverse mal posé. Le fait de trouver une solution ne suffit pas, il faut aussi que cette solution soit unique et qu'elle dépende de façon continue des données initiales au problème. C'est ainsi que Hadamard[5] a défini les propriétés de modélisation mathématique d'un phénomène physique. Un problème mal posé se crée à cause d'un manque d'information. Nous observons un phénomène et nous essayons de le cerner sans maîtriser les autres éléments entourant celui-là. Dans le domaine de la vision par ordinateur, il existe plusieurs autres problématiques mal posées du fait que nous partons de compréhensions et de perceptions en trois dimensions et nous voulons les reproduire dans un univers en deux dimensions, l'image. Nous pouvons citer comme exemples de problèmes mal posés la restauration d'image, le *matting*, *l'inpainting*, etc.

4.3.2 DEFINITION

La première modélisation et définition des contours actifs a été introduite par Kass et al. en 1988 [Kass et al. 1988]. Les auteurs ont proposé une définition mathématique de ce modèle, aussi connu sous le nom de *snake* en référence aux possibilités de déformation de la courbe. Dans cette section, nous

[5] http://fr.wikipedia.org/wiki/Jacques_Hadamard

allons présenter la définition initiale du modèle des contours actifs, telle que présenté par Kass et al.. Par la suite nous allons décrire les évolutions clé que cette approche a vues et qui ont contribué à son succès par la présentation de certaines implémentations qui ont marqué son évolution.

4.3.2.1 LE MODELE CLASSIQUE

Un contour est représenté de façon paramétrique sur un plan par une courbe *C(s)*, avec *s* son abscisse curviligne. A tout point de *C* on peut associer sa représentation C(s), Figure 46:

$$C(s) = \begin{bmatrix} x(s) \\ y(s) \end{bmatrix}, \quad avec\ s \in [0, S] \tag{12}$$

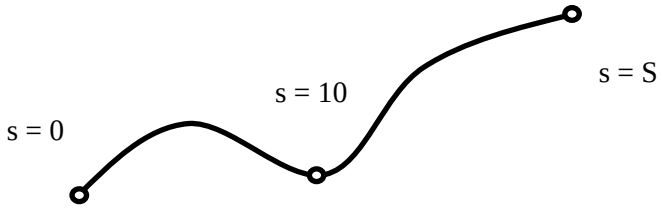


FIGURE 46 : COURBE PARRAMETRIQUE

Dans la littérature, on fait souvent référence au cas où $C(0) = C(S)$ car la plupart des modélisations et des problématiques sont liées à des problématiques relatives à la recherche d'un contour fermé. Dans le cas où le contour représente une courbe ouverte sa représentation paramétrique reste la même en considérant que S correspond à l'une des extrémités de cette courbe. L'utilisation d'une représentation paramétrique permet de dépasser les problèmes posés par une correspondance entre abscisses et ordonnées qui ne serait pas univoque. Cette représentation paramétrique du contour actif peut être définie dans un domaine normalisé par :

$$C \coloneqq [0,1] \rightarrow R^2 \tag{13}$$

Le domaine de définition de la courbe est normalisé à [0,1] et l'abscisse curviligne est liée au paramètre de la représentation.

La courbe associée au contour actif, non nécessairement fermée, est initialement placée dans la zone d'intérêt de l'image. Selon les auteurs originaux [Kass et al. 1988], en rapport avec leur cas d'utilisation, la courbe doit se trouver autour de l'objet « d'intérêt ». Le critère à optimiser, dont *C* est

une solution, est une somme pondérée d'énergies qui traduisent les forces de natures différentes qui définissent, chacune par leurs contradictions ou par leurs complémentarités, les contraintes du problème à résoudre.

$$E(C) = \int_0^1 E_1(C(s)) + E_2(C(s)) + \cdots + E_n(C(s))\, ds \tag{14}$$

Il existe trois principales familles d'énergies utilisées dans les modèles des contours actifs selon [Rousselle 2003]: énergies associées aux forces internes issues des caractéristiques de la courbe elle-même, les énergies relatives aux forces externes issues de l'attache aux données, un comportement prédéfini et les forces de contexte que l'on peut intégrer en tant que forces externes. Le contour actif s'équilibre et la convergence est atteinte lorsque l'énergie à optimiser est minimale dans l'espace qui l'entoure.

$$\int_0^1 E_{int} + \lambda E_{ext} \tag{15}$$

L'évolution d'un contour d'une position à une autre dans l'espace 'image' peut être considérée comme une évolution sur un axe temporel sous la contrainte d'un ensemble de forces. Cette évolution peut se formaliser mathématiquement sous la forme d'une équation exprimant la vitesse de déplacement du contour ou plus particulièrement de ses points. Il existe plusieurs manières d'obtenir une équation d'évolution. L'une, celle qui a été introduite par Kass et al, consiste à dériver cette équation de la minimisation d'une fonctionnelle d'énergie, on parle alors d'approche variationnelle. On utilise l'équation d'Euler-Lagrange pour résoudre le problème de minimisation de l'énergie, pour définir le contour actif en terme d'énergie, ainsi l'équation d'énergie, selon [Kass et al. 1988], s'écrit:

$$E(C) = \underbrace{\alpha \int_0^1 |C'(s)|^2\, ds + \beta \int_0^1 |C''(s)|^2\, ds}_{\text{Energie interne}} + \underbrace{\gamma \int_0^1 g^2 |\nabla \mathrm{I}(C(s))|^2\, ds}_{\text{Energie externe}} \tag{16}$$

Avec C', la dérivé première de C et C'' sa dérivé seconde. g est une gaussienne, vérifiant g(0)=1 et $\lim_{s\to+\infty} g(s) = 0$. α, β et γ sont des constantes positives.

Un minimum doit respecter les équations d'Euler-Lagrange :

$$\begin{cases} -(\alpha C') + (\beta C'') + \nabla(g * \nabla \mathrm{I}) = 0 \\ \exists\, k \in \mathbb{R}, \qquad \alpha s - \vartheta = k\sqrt{1 + \dot{s}^2} \end{cases} \tag{17}$$

Cette équation peut avoir plusieurs solutions puisque l'énergie peut avoir plusieurs minimums locaux. La solution que l'on cherche est localisée dans une région donnée et on suppose qu'elle possède une valeur approchée de la solution C^*. Cette implémentation est la plus courante. Elle est la plus utilisée et la plus déclinée. Elle nécessite pour atteindre la solution, à chaque itération, des inversions de la matrice, ainsi que le réglage du coefficient d'évolution ϑ.

La fonctionnelle d'énergie peut être schématisée comme la somme de deux classes de termes énergétiques. Pour mieux expliquer le fonctionnement de cette approche nous pouvons faire une analogie avec la physique ainsi, comme nous l'avons présenté dans la Figure 47 et la Figure 48, l'ensemble des énergies représente en effet, un ensemble de forces, agissant sur le système, qui est le contour actif. Les forces intrinsèques gèrent la cohésion de la courbe et les forces extrinsèques traduisent à leur tour un ensemble de contraintes agissant sur le système.

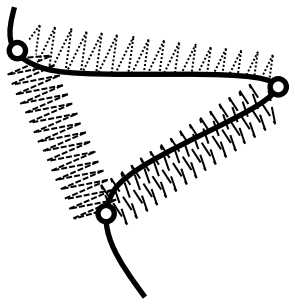


FIGURE 47 : RESORTS REPRESENTANT LES FORCES INTRINSEQUES ET MONTRANT LEURS IMPACTS SUR LA COURBURE ET L'ECARTEMENT DES POINTS

La première classe, représentée par les deux premiers termes de l'équation (16), comme nous l'avons déjà remarqué, concerne l'énergie interne du contour visant à contrôler des contraintes intrinsèques à celui-ci, comme par exemple sa régularité, l'élasticité, etc. La deuxième classe est relative au terme d'attache aux données, représenté par le troisième terme de l'équation (16), faisant interagir le contour actif avec des caractéristiques extraites de l'image.

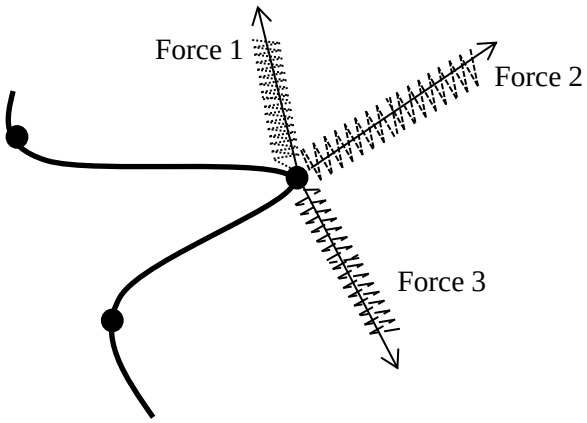


FIGURE 48 : RESSORTS MODELISANT L'AGISSEMENT DES FORCES EXTRINSEQUES SUR UN POINT DE LA COURBE

Cette approche est toutefois limitée par plusieurs inconvénients dont le problème du coût de calcul qui est relativement important (inversion de matrice dans le cas de l'approche variationnelle). Amini et al. [Amini et al. 1990] ont signalé que la méthode pouvait s'avérer numériquement instable et que les points avaient tendance à s'entasser sur certaines portions du contour obtenu par extraction de la carte des contours (carte des forts gradients). La courbe représentant le contour doit être fermée. Cette approche nécessite que l'initialisation soit assez proche de la solution finale. Aussi, la définition de fonctionnelles d'énergies est assez spécifique. Elle dépend fortement de la modélisation que l'on peut faire de l'environnement et des caractéristiques du contour recherché. Il s'agit d'introduire le plus de connaissance possible de manière à limiter les minimums locaux au voisinage de la solution recherchée.

Dans les travaux de [Foulonneau 2004], le terme d'énergie externe est aussi appelé critère. Il est construit à partir de descripteurs. Un descripteur est une mesure faite sur l'image permettant de caractériser une frontière ou une région. Un descripteur de frontière serait par exemple la carte des gradients de l'image, un descripteur d'une région R pourrait être la moyenne des niveaux de gris des pixels de l'image inclus dans R. Selon le choix du descripteur adopté dans la fonctionnelle d'énergie, on dérivera différents types de contours actifs plus ou moins performants en fonction de la difficulté de l'image à analyser (ex : image bruitée, image non bruitée, etc.).

Une autre manière d'approcher le problème peut se faire en considérant davantage d'éléments et sans se limiter à étudier les descripteurs contour ou les descripteurs région. On peut tirer l'avantage des deux à la fois, c'est l'approche géométrique. Ici, le principe du contour actif consiste à bien dégager

une interprétation physique au problème en définissant un ensemble de forces relatives à n'importe quel descripteur et par la suite construire une équation d'évolution. La difficulté est alors dans le choix d'un ensemble de forces pouvant correctement et mutuellement représenter le système.

Dans l'espace discrétisé de l'image, un contour actif peut être représenté par N sites, ce qui nous permet de lui associer une représentation par un ensemble ordonné de sommets, $V = [v_1, v_2, \dots, v_n]$ ayant pour chaque $v_i \in V$, en plus de leur emplacement sur la courbe selon le paramètre, des coordonnées (x, y) dans l'espace image.

Ainsi l'équation d'énergie globale peut s'écrire de la manière suivante :

$$E = \sum_{v_k \in V} E(v_k) \tag{18}$$

avec

$$E(v_k) = \sum_{i=1}^{n} \lambda_i E_i(v_k) \tag{19}$$

Avec $E(v_k)$ l'énergie relative au $k^{éme}$ point de la courbe. La réécriture de l'équation de l'énergie globale sous la forme ci-dessus, nous permet d'introduire d'autres implémentations plus optimisées. Dans la section suivante, nous allons détailler ces implémentations qui représentent une variante de l'autre.

4.3.2.2 RESOLUTION PAR PROGRAMMATION DYNAMIQUE

Dans l'optique de corriger certains problèmes relatifs à la modélisation de Kass et al. et en considérant que le problème possède la propriété de sous structure optimale, Amini, Weymouth et Jain [Amini et al. 1990] ont introduit l'utilisation de la programmation dynamique pour trouver une solution optimale. C'est un ensemble de méthodes d'optimisation pour répondre à un problème de planification posé lorsque des décisions doivent intervenir à des périodes discrètes et qu'à chaque période, un nombre fini d'options de décisions peuvent être prises. En partant d'un premier point du contour, il est possible de traiter le problème global de minimisation comme un problème de minimisation qui, pour chaque ensemble fini d'étapes, prend une décision parmi un ensemble fini de solutions possibles. Le problème se ramène alors à un problème d'optimisation d'une fonction numérique de plusieurs

variables. La formulation standard de la programmation dynamique peut s'écrire, donc, de la manière suivante:

$$S_i(v_{i+1}, v_i) = \min_{v_{i-1}} \left\{ S_{i-1}(v_i, v_{i-1}) + \sum_{k=1}^{n} \lambda_i E_i\,(v_k) \right\} \tag{20}$$

$S_i(v_{i+1}, v_i)$ représente le potentiel de l'énergie globale sur la portion bornée par les deux points de la courbe v_i et v_{i+1} qui sont deux points successifs. λ_i est le poids prédéfini associé à l'énergie E_i.

La convergence de cette approche est garantie, pour un voisinage de taille m, et un contour de n points, la complexité est de $O(nm^3)$.

4.3.2.3 RESOLUTION PAR L'APPROCHE 'GREEDY'

L'utilisation de l'algorithme *greedy* pour minimiser l'énergie des contours actifs est une solution très intéressante. Dans la même optique d'amélioration du modèle proposé par Kass et al., D.J. Williams et M. Shah ont développé une nouvelle approche de résolution qui a été appelée '*greedy snake*' [Williams and Shah 1992]. Cette alternative est devenue assez fréquente par rapport à l'approche variationnelle. Parmi les problèmes évoqués dans [Williams and Shah 1992] et auxquels leur proposition apporte une solution nous pouvons citer le problème de la stabilité numérique des approches variationnelle, déjà évoquée par Amini et al.. Aussi, la complexité de l'algorithme a été significativement réduite. Williams et Shah, discrétisent l'expression d'énergie présenté dans l'approche classique par [Kass et al. 1988] en utilisant des différences finies et procèdent successivement par une recherche locale (Figure 49). Par exemple, pour l'énergie d'uniformité présenté par $\int_0^1 |C'(s)|^2 \, ds$, ils utilisent la formulation suivante:

$$\left\| \frac{dv_i}{ds} \right\|^2 = \|v_i - v_{i-1}\| = (x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 \tag{21}$$

L'approche *greedy* est plus rapide que les deux approches précédentes. À chaque itération, l'algorithme possède une complexité de $O(nm)$, au lieu de $O(nm^3)$ nécessaire pour l'approche par programmation dynamique, en considérant un voisinage de taille *m*.

Nous observons donc différentes catégories de contours actifs. Celles-ci se différencient par la façon avec laquelle on déduit l'équation d'évolution, par le mode de représentation du contour actif, et finalement par le terme d'attache aux données qui peut être soit basé sur les frontières, les régions,

hybride ou bien qui peut être déduit d'une connaissance sur les données et le problème en question. Nous verrons dans la section 4.6 comment insérer des contraintes externes spécifiques dérivées de la connaissance *a priori* que l'on a sur l'objet à identifier ou plutôt sur notre problématique et l'objectif à atteindre dans l'image.

Notre choix, bien sûr pour ces raisons d'efficacité et de modularité s'est porté sur cette optimisation par algorithme *greedy*. Dans le reste de ce chapitre, nous nous restreignons au cas des contours actifs bidimensionnels évoluant dans le plan 2D, pour un état de l'art exhaustif sur les contours actifs nous conseillons les travaux de [Gastaud 2005]

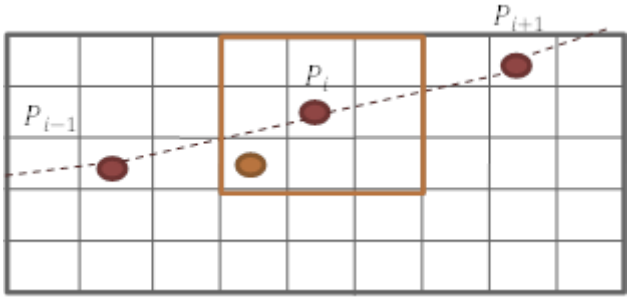


FIGURE 49 : RECHERCHE LOCALE DU MINIMUM, P_i EST POINT DE LA COURBE, EN ORANGE LA NOUVELLE POSITION DE P_i APRES UNE RECHERCHE LOCALE DU MINIMUM.

4.4 PROPAGATION DE COURBE

Pour suivre une courbe tracée manuellement sur une image, il s'agit de traquer un ensemble de points. Les méthodes classiques d'estimation de mouvement [Baker and Matthews 2004], ne permettent pas la propagation de ces points sur un nombre important d'images consécutives. D'autres méthodes plus récentes et plus avancées existent, tel que [Irani 2002], permettent de propager des points d'intérêt sur un grand nombre de frames (bien sûr proportionnellement à la complexité de la scène et à la vitesse du mouvement à suivre). Les auteurs expliquent que le choix des points à suivre constitue une étape primordiale dans le succès de leur approche et ce choix nécessite une phase de définition assez spécifique. Dans le cadre de la propagation de courbe, nous allons voir quelles sont les spécificités des points des courbes utilisées pour la désignation ou le marquage d'objets vidéo et pourquoi les méthodes de suivi ponctuel ne sont pas efficaces dans notre contexte. Nous analyserons quelles sont les difficultés principales auxquelles nous avons essayé de répondre par notre proposition.

4.4.1 L'INITIALISATION ET LE MARQUAGE D'OBJETS VIDEO

L'initialisation de la courbe est l'une des difficultés majeures du problème des contours actif. Cette étape contribue à l'accélération de la convergence de l'algorithme. Les résultats obtenus, leurs qualités et le temps de recherche et d'exécution lui sont, aussi, liés. Un problème de contour actif peut être vu comme un problème de recherche de position optimale (position des points composant le contour). Une recherche efficace est un problème complexe d'ordre O((MxN) ×|V|), avec MxN, la taille de l'image et V l'ensemble de points formant la courbe (ou le contour). A partir d'une courbe ou un contour initial, si l'on s'autorise de déplacer chaque point du contour d'un pixel, la complexité de la recherche opérationnelle tombe à O(9×|V|). Un tel coût est inévitable, il faut donc avoir une idée du résultat final (la position où la courbe va se stabiliser) et savoir placer le contour initialement le plus proche de ce résultat afin de réduire le nombre d'itérations. Si nous considérons que le coût de la recherche est de O(1×|V|), le fait que la courbe initiale soit éloignée de l'objectif souhaité implique qu'il nous faut plus d'itérations pour atteindre le résultat, 100 itérations par exemple. Dans ce cas le coût total pour que l'algorithme converge est O(l00×|V|). Pour optimiser l'initialisation, il existe certaines approches permettant d'optimiser cette phase par l'emploi de techniques d'apprentissage.

Dans notre contexte nous allons uniquement nous intéresser à l'initialisation qui requiert l'intervention de l'utilisateur. La nature interactive des applications dans lesquelles nous nous positionnons fait qu'un processus d'initialisation automatisé n'est pas vraiment envisageable. En effet la variété des objets se trouvant dans une vidéo et qu'un utilisateur voudra sélectionner ne nous permet pas de nous investir dans un processus basé sur l'apprentissage (variation des formes, des couleurs, des textures, etc.). Et même au sein d'une famille unique d'objets, du fait que nous nous adressons à des objets de nature déformable, il est possible, au mieux, de placer automatiquement des marquages via un processus de classification [Viola and Jones 2001] couplé à une phase d'apprentissage, et donc la/les courbes, à l'extérieur de l'objet en prenant l'extérieur de la boîte englobant la zone dans laquelle l'objet se situe.

FIGURE 50 : PROPAGATION FRAME PAR FRAME

Les marqueurs, dans notre contexte sont rentrés manuellement par l'utilisateur, ces courbes sont utilisées pour désigner l'objet d'intérêt. Une courbe placée à l'intérieur de l'objet est propagée par notre algorithme pour fournir un marquage pouvant marquer au mieux l'objet dans la frame suivante (Figure 50). Ce résultat, lui-même, est utilisé par la suite pour servir d'initialisation dans l'image suivante. Ce processus peut être ainsi effectué continuellement ou successivement jusqu'à ce qu'il devienne erroné (le marquage ne désigne plus correctement l'objet sélectionné initialement par l'utilisateur, par exemple) ou jusqu'à la fin la séquence vidéo ou jusqu'à la disparition de l'objet.

4.4.2 POINTS CARACTERISTIQUES ET POINTS DU MARQUAGE

Pour être correctement suivis, les points formant la courbe dessinée par l'utilisateur doivent avoir un certain nombre de propriétés spécifiques, par exemple ils doivent être des points saillants (des points d'intérêt) comme ceux définis dans [Rav-Acha and Peleg 2006; Shi and Tomasi 1994]. Image après image, les erreurs de suivi de chaque point sont propagées et accumulées progressivement. Si la position relative issue par le mouvement estimé est erronée, la propager de nouveau ne fera qu'augmenter les erreurs. Cela arrive souvent même quand les points à suivre sont plutôt bien choisis. Les méthodes ponctuelles de suivi sont très sensibles à la ressemblance de texture et au problème d'ouverture (qui est issu de l'hypothèse de conservation de l'intensité lumineuse ou de la luminance, Figure 51). Le problème qui se pose en plus dans notre cas est que les points ne sont pas choisis, ils sont issus d'un traçage fait par l'utilisateur, ce qui rend leur suivi beaucoup plus sujet aux erreurs.

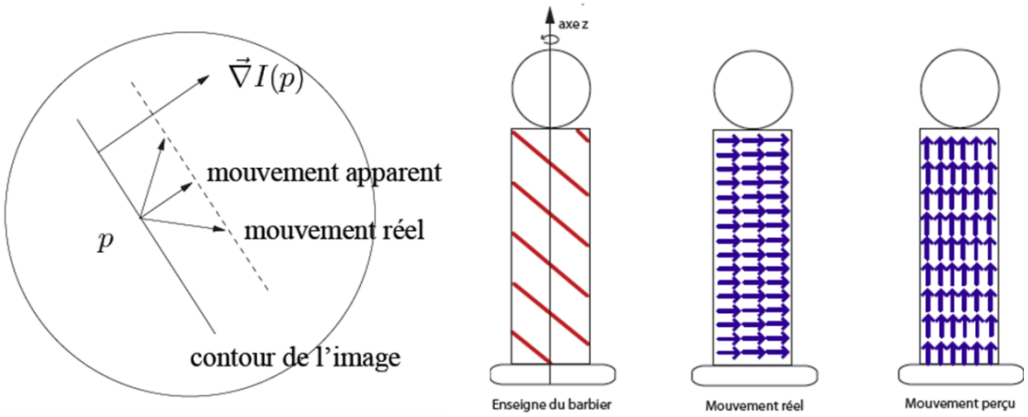


FIGURE 51 : ILLUSTRATION DU PROBLEME D'OUVERTURE, SEULE LA COMPOSANTE NORMALE AU DEPLACEMENT EST MESURABLE [Louvat 2008]

En réalité les points de la courbe ne sont pas géométriquement indépendants. Les traiter séparément ne fera qu'ajouter de l'incohérence au suivi. Il nous a donc semblé indispensable de conserver sa cohérence à cet ensemble de points sélectionnés par l'utilisateur et de traiter cet ensemble de points comme un tout, comme une courbe. L'utilisation des courbes actives répond parfaitement à ces contraintes. En effet, grâce aux forces internes qui gèrent la cohésion des points sous la forme d'une courbe entière, où le déplacement d'un point, relativement à une contrainte externe, impacte la situation globale et peut engendrer un repositionnement de ses points voisins.

Dans la section suivante nous allons aborder une étape clé dans notre démarche de propagation. Cette étape consiste en la compréhension et la définition des éléments et contraintes permettant à la courbe (le marquage) de respecter notre objectif initialement défini qui consiste à faire évoluer la courbe tout en continuant à designer l'objet initialement désigné par l'utilisateur.

4.4.3 PROBLEMATIQUES DE LA PROPAGATION DE COURBE

Quelle est la fonctionnalité de la courbe que l'on veut propager ? Cette question se pose implicitement à l'utilisateur qui veut désigner des zones. La réponse à cette question aidera à déterminer les propriétés qui permettent de propager la courbe tout en lui conservant sa fonctionnalité. Nous nous plaçons dans l'hypothèse où l'utilisateur trace une courbe pour désigner un objet dans la scène. La courbe passe donc par un certain nombre de régions de textures et homogénéités différentes. Ces régions composant l'objet, sont caractérisées par différents types d'homogénéités. La courbe dessinée

est souvent placée plutôt dans la partie médiane de ces zones comme on le voit sur la Figure 52 et la Figure 52-A. En effet la précision de l'utilisateur est faible, il est alors plus sûr pour lui de dessiner la courbe loin des bords de l'objet. La Figure 52 montre ce comportement sur un échantillon de personnes à qui nous avons demandé de dessiner une courbe via laquelle ils désigneraient un objet qui leur est indiqué.

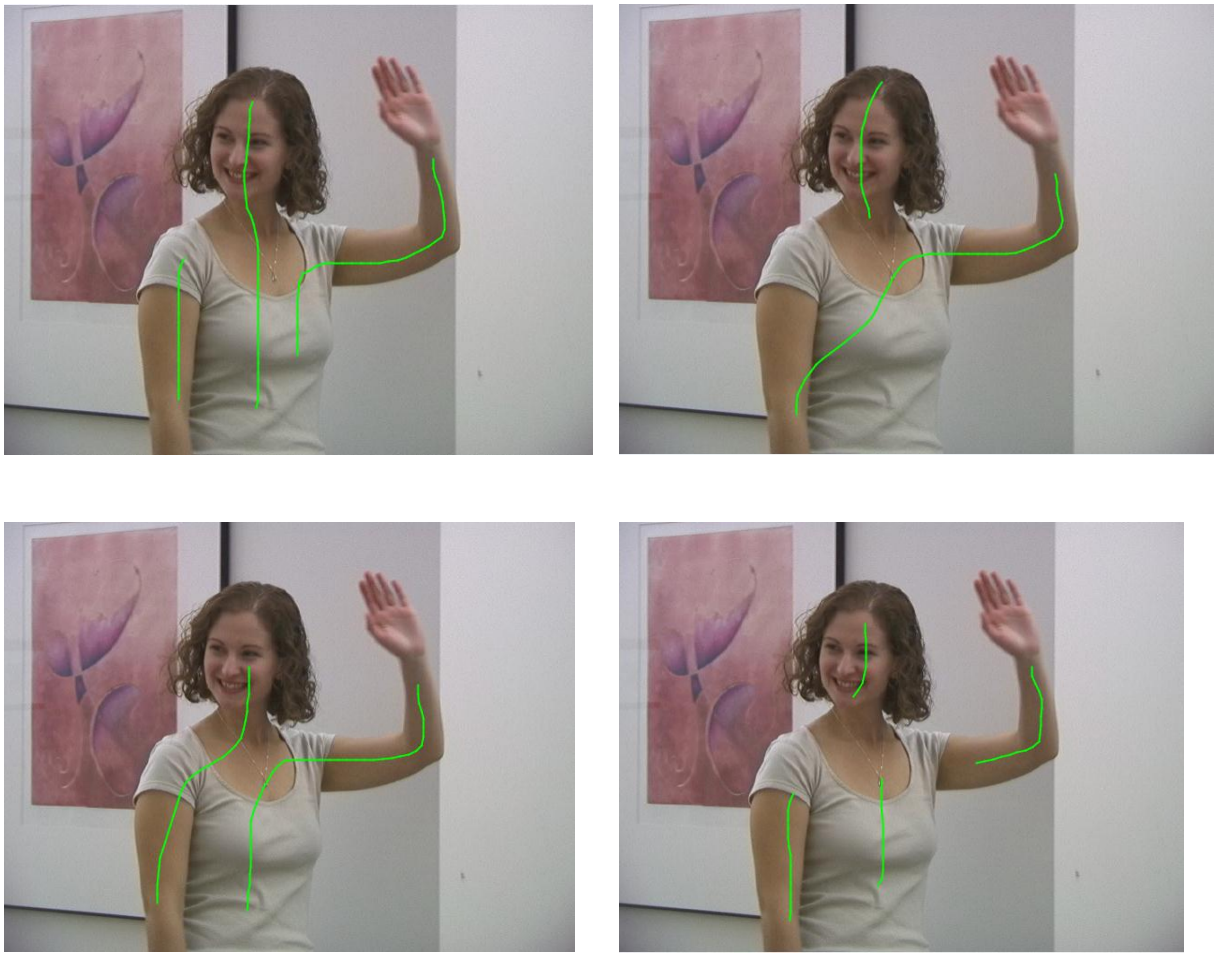


FIGURE 52 : COURBES DESSINEES PAR DIFFERENTES PERSONNES POUR INDIQUQER LA FILLE PRESENTE DANS L'IMAGE

Pour désigner le même objet dans les images suivantes de la vidéo, on souhaite que la courbe reste dans les zones considérées et se positionne le plus possible au milieu de celles-ci de manière à limiter la dérive. La courbe n'est pas caractérisée par un fort gradient. Il ne s'agit pas ici, comme c'est le cas dans de nombreuses études, d'une courbe de contour qui limite un objet que l'on suit mais plutôt d'une courbe qui va désigner les différentes parties d'un objet. Nous avons identifié trois contraintes qui

doivent être respectées lors de la propagation d'une courbe tracée sur une image à l'instant *t* vers l'image à l'instant *t+1*:

- Si l'objet désigné se déplace, la courbe doit être déplacée en conséquence dans l'image.
- Il faut conserver les caractéristiques des zones traversées, en tenant néanmoins compte du fait qu'une zone peut subir des changements aussi bien au niveau de la forme qu'au niveau de l'illumination.
- Pour prendre moins de risque, la courbe doit se déplacer vers les zones les plus homogènes, localement, pour minimiser les erreurs et essayer d'approcher la région médiane de chaque zone.

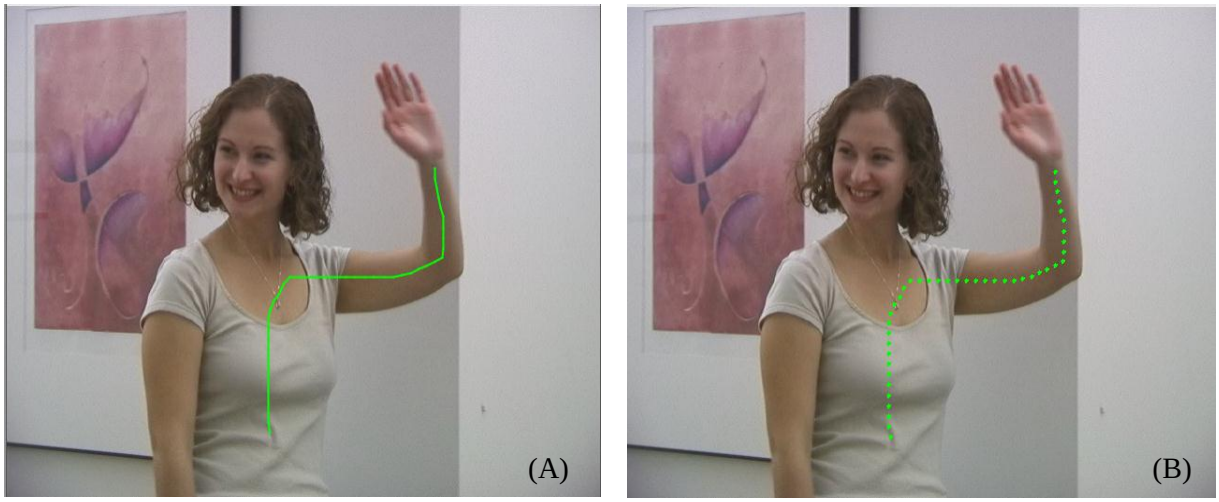


FIGURE 53: (A) EXEMPLE D'UN GRIBOUILLIS DESSINE PAR L'UTILISATEUR. (B) LE GRIBOUILLIS DISCRETISE EN UN ENSEMBLE DE POINTS.

Dans notre cas, on ne peut pas avancer d'hypothèses sur la nature de l'objet marqué par la courbe dessinée par l'utilisateur. Les modèles des courbes de Bézier ou les Splines, par exemple, ne sont pas générées de façon assez souple pour modéliser les déformations nécessaires pour répondre aux trois contraintes exprimées ci-dessus.

Pour l'ensemble des régions composant l'objet, nous proposons de considérer la courbe globalement et de modéliser les forces qui conduisent à la repositionner au long de la vidéo.

Pour assurer une propagation cohérente, il est nécessaire d'introduire ces nouvelles contraintes dans la gestion de l'évolution de la courbe. Nous avons essayé de voir ces contraintes sous forme de forces et

nous avons défini les fonctionnelles d'énergies associées. Cela nous a permis de propager et de repositionner la courbe sur les images suivantes. L'estimation de la position de la courbe d'une image à l'autre se fait via un processus d'optimisation, de minimisation d'énergie. La position optimale dans l'image suivante est alors calculée par programmation dynamique en utilisant l'algorithme Greedy [Williams and Shah 1992].

Afin d'assurer une propagation cohérente en tenant compte du marquage C et de l'objet marqué, l'ensemble de points constituant un gribouillis désignant un objet composé de différentes régions, est soumis à un ensemble de forces qui se décompose en deux familles, les unes associées à la courbe elle-même et les autres liées aux données. Cet ensemble de forces correspond à l'intégration des différentes contraintes citées précédemment. Notre objectif est de formuler ces forces en utilisant une modélisation énergétique et par la suite de minimiser la somme des énergies correspondantes pour aboutir à une solution qui minimise les erreurs et qui est optimale pour l'ensemble de ces forces réunies.

$$\mathrm{E_{global}(C)} = \mathrm{E_{int}(C)} + \mathrm{E_{ext}(C)} \tag{22}$$

Même si nous sommes en train de présenter la solution comme étant un minimum global, les forces agissent, en effet, localement sur les différents points de la courbe C. L'équilibre de cet ensemble de force est lui en effet obtenu globalement sur la courbe en tenant compte de l'optimisation issue de chaque point. Ainsi, si nous discrétisons la courbe C par un ensemble ordonné de N points, l'énergie $\mathrm{E_{global}(C)}$ peut être estimée par la formule suivante:

$$\hat{\mathrm{E}}_{\mathrm{global}}(\mathrm{C}) = \sum_{\mathrm{p}\in\mathrm{R}} \mathrm{E_{int}(p)} + \mathrm{E_{ext}(p)} \tag{23}$$

Avec $\mathrm{R} = \{p_1, \dots, p_N\}$ un ensemble de points représentant la nouvelle courbe optimale.

Nous allons, dans la section suivante, présenter les deux ensembles de forces internes et externes que nous avons définis.

4.4.4 MODELISATIONS DES CONTRAINTES

Dans cette section nous allons voir comment nous avons procédé pour concrétiser les contraintes issues de notre problématique et que nous avons décrites précédemment. Les contraintes intrinsèques à la courbe et les contraintes relatives à l'attache aux données seront explicitées.

4.4.4.1 FORCES INTERNES

Les forces internes sont là pour guider le comportement global de la courbe. Comme présenté dans [Kass et al. 1988], nous utilisons une force relative à la continuité et une autre relative à la courbure. L'énergie représentant la force de continuité va essayer de maintenir la cohésion des points et donc va uniformiser les distances séparant les couples de points successifs. L'énergie relative à la force de courbure va avoir une influence sur la rigidité de la courbe. Ces deux fonctionnelles d'énergie vont influencer la forme de la courbe au fur et à mesure de son évolution. Des poids $\omega_1$et ω_2 sont utilisés pour privilégier un comportement ou un autre. Plus ω_1 est élevé plus les points de la courbe auront tendance à se rapprocher au fur et à mesure des itérations et plus ω_2 est élevé plus les points qui ne sont pas des voisins directs auront tendance à se rapprocher et donc plus la courbe se courbera.

Uniformité

L'énergie relative à l'uniformité de la courbe va essayer de standardiser les écarts entre les points de celle-ci. Notons avg_{dist} la distance moyenne séparant deux points successifs de la courbe, points p_i et p_{i+1}. Pour chaque point p_i de la courbe, la fonction définissant l'énergie relative à la force d'uniformité, dans un voisinage de p_i, s'écrit comme suit :

$$E_{unif}(p) = |avg_{dist} - \|p\, p_i\|| \tag{24}$$

Courbure

Soit p_{i-1}, p et p_{i+1} trois points successifs de la courbe. Définissons u_x comme étant le projeté orthogonal de p_{i-1}, p sur l'axe des abscisses. Soit v_x la longueur du projeté orthogonal de p, p_{i+1} sur le même axe. Soit u_y et v_y, respectivement les longueurs des projections équivalentes à u_x et v_x mais sur l'axe des ordonnées, (Figure 54). Pour chaque point p_i de la courbe, nous définissons, dans un voisinage de p_i, l'énergie relative à la force de courbure comme suit :

$$E_{courb}(p) = \left(\frac{(u_x + v_x)}{\|p_{i-1}, p\|\|p, p_{i+1}\|}\right)^2 + \left(\frac{(u_y + v_y)}{\|p_{i-1}, p\|\|p, p_{i+1}\|}\right)^2 \quad (25)$$

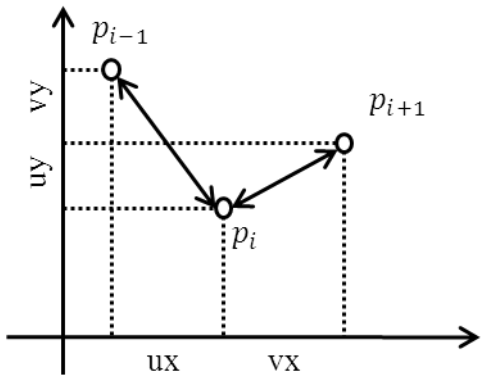


FIGURE 54 : MESURE DE COURBURE

D'où, chaque point p de la courbe contribue à l'énergie interne totale par la quantité :

$$\mathrm{E_{interne}(p)} = \omega_1 \mathrm{E_{unif}(p)} + \omega_2 \mathrm{E_{courb}(p)} \quad (26)$$

4.4.4.2 FORCES EXTERNES

La définition des forces externes est plus sensible. Elle est directement reliée à la nature du problème et à la manière dont nous l'avons modélisé. Désigner un objet via un ensemble de marquages (gribouillis), consiste à la propagation des gribouillis dessinés initialement par l'utilisateur sur la première frame de la séquence vidéo. Ceci peut être représenté, donc, par la propagation d'un ensemble de points issus de la courbe du gribouillis désignant un objet de haut niveau. Nous avons décidé de modéliser les contraintes analysées précédemment par un ensemble de trois forces :

- Une relative à l'estimation du mouvement en chaque point de la courbe. Le suivi du déplacement est essentiel. (C1)
- Une autre relative à la similarité des couleurs. L'assurance de l'homogénéité de la propagation est importante. (C2)

- Une dernière qui met en évidence une direction à privilégier pour réduire les erreurs de positionnement lors de la propagation. La garantie de la stabilité du déplacement de la courbe est primordiale. (C3)

La composition des trois contraintes C1, C2 et C3 assure au système un équilibre et une garantie le permettant d'aboutir à une convergence représentant la solution la plus adéquate en considérant notre problème.

Mouvement

Notre étude n'est pas limitée aux objets rigides, le mouvement d'un objet déformable n'est pas identique pour toutes ses parties. Quand l'objet bouge, son mouvement global peut être constitué de différents mouvements associés à différents éléments qui le composent. Estimer le mouvement localement au niveau des points est nécessaire pour assurer que la courbe puisse coller dans sa globalité au mouvement de l'objet. Notons p' l'image estimée par le flux optique du point p de la courbe dans I_t à l'instant *t*. Nous noterons p une éventuelle position de la courbe à *t+1*. La première force, que nous associons à la contrainte C1, va tenter de minimiser la distance euclidienne entre p et p'_t, pour obtenir la nouvelle position p_{t+1}dans l'image I_{t+1}. Ainsi on note sa fonction d'énergie associée :

$$E_{mvt}(p) = |p\ p'| \tag{27}$$

Cette énergie assurera pour chaque point le fait qu'il sera attiré vers sa position estimée suite au déplacement de l'objet.

Texture

Entre les images I_t et I_{t+1}, un point de la courbe doit rester dans la même zone de l'objet. Une zone peut être caractérisée par sa couleur ou aussi par sa texture. La caractérisation par la texture est plus coûteuse en temps de traitement et de calcul, nous allons alors limiter notre étude à une couleur moyenne sur un voisinage de chaque point, nous noterons cm_p la couleur moyenne autour du point p calculée sur une fenêtre de voisinage 3x3, cela a pour effet de rendre les comparaisons de couleur un peu moins strictes. Nous avons utilisé le CIE Lab comme espace couleur. Ce dernier est plus adapté que l'espace RVB, en effet il nous permet la distinction entre les composantes chromatiques et la luminance. Cette deuxième force doit pousser alors un point p_t de la courbe à évoluer vers une nouvelle position p, dans I_{t+1}, ayant des couleurs similaires en respectant une tolérance par rapport au

changement d'illumination. L'énergie correspondant à la deuxième contrainte C2 est, ainsi, calculée en se rapportant à une distance euclidienne pondérée dans l'espace couleur comme suit:

$$E_{couleur}(p) = \left(cm_{pt}(a) - cm_p(a)\right)^2 + \left(cm_{pt}(b) - cm_p(b)\right)^2 + lw\ \left(cm_{pt}(l) - cm_p(l)\right)^2 \quad (28)$$

Le choix du poids *lw* a été fixé de manière empirique et nous avons pris la valeur de $1/4$ pour réduire l'impact du changement de luminosité.

Stabilité

La dernière force, issue de la contrainte C3, va essayer de reproduire la tendance prudente de l'utilisateur. Comme il manque de précision lorsqu'il dessine, l'utilisateur est amené à éviter les bords de l'objet qu'il veut désigner (ce qui lui évite de sortir de l'objet et lui évite aussi de passer beaucoup de temps à tracer son marquage). La force que nous introduisons a pour but de pousser la courbe à rester au milieu des différentes zones de l'objet. Cette force pousse les points vers les endroits de plus grande homogénéité. Cela augmente la stabilité de la propagation.

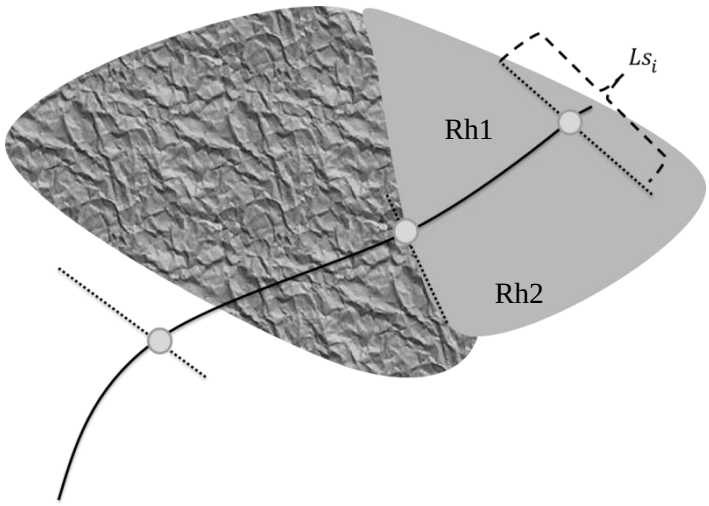


FIGURE 55: REGIONS TRAVERSEES PAR UNE COURBE (B2) EST MOINS AFFECTE PAR LES MOVEMENT DE L'OBJET QUE (B2) QUI EST PLUS REPRESENTATIF DE LA REGION (R1) QUE (B1).

Relativement à la Figure 55 si l'on note Rh la zone homogène, au sens de la couleur, à laquelle appartient pt, elle est partagée, au moins localement en deux zones Rh1 et Rh2 par la courbe. La

largeur dp d'une de ces zones est mesurée en p par la longueur du segment orthogonal à la courbe en pt entièrement contenu dans la zone. La zonede la plus profonde est celle où les risques de sortir de l'objet sont les moins grands. Nous notons p_{lt} l'extrémité du segment qui est la plus éloignée de p_t et p'_{lt} son image dans la frame *t+1*. Notre objectif, en partant de l'état d'un point à l'instant *t* est de privilégier la propagation dans une direction par rapport aux autres. Nous illustrons l'effet sur la Figure 56. La fonction d'énergie correspondante à minimiser s'écrit alors:

$$E_{stabilité}(p) = |p\ p'_{lt}| \tag{29}$$

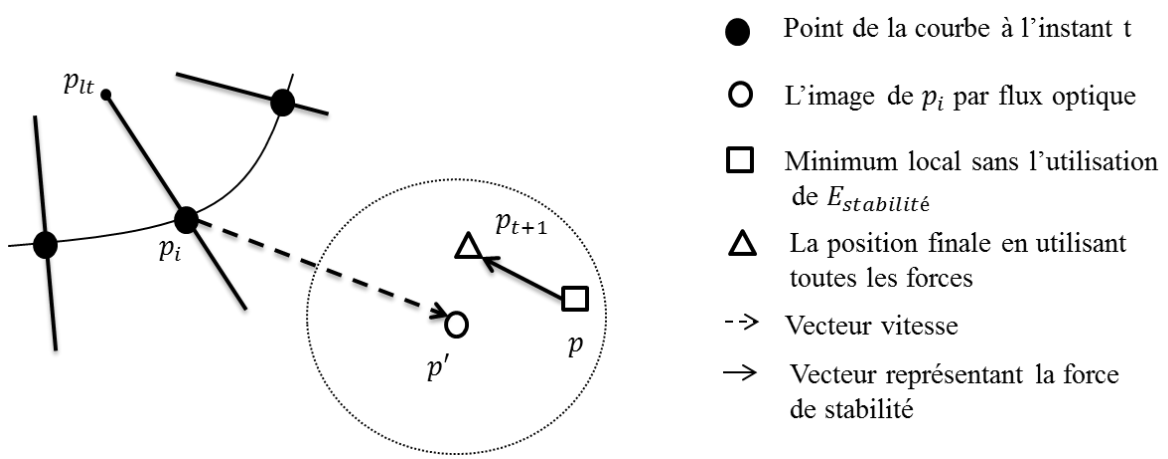


FIGURE 56 : ILLUSTRATION DE LA FORCE DE STABILITE

D'un point de vue pratique, nous détectons les régions homogènes aux alentours de la courbe en nous basant sur la couleur des segments orthogonaux à la courbe en chaque point (Figure 56, Figure 57). Pour calculer l'énergie relative à cette force, nous estimons la normale à la courbe en chaque point p_t, ensuite nous calculons sur cette droite un segment de similarité. Le segment de similarité associé à chaque point p_t est le segment de longueur maximum de couleur uniforme dans cette direction. Ce qui veut dire que tous ses points ont une couleur similaire à celle de p_t, et il est maximum au sens de la longueur. Les extrémités de ce segment sont les points limite de la tolérance couleur.

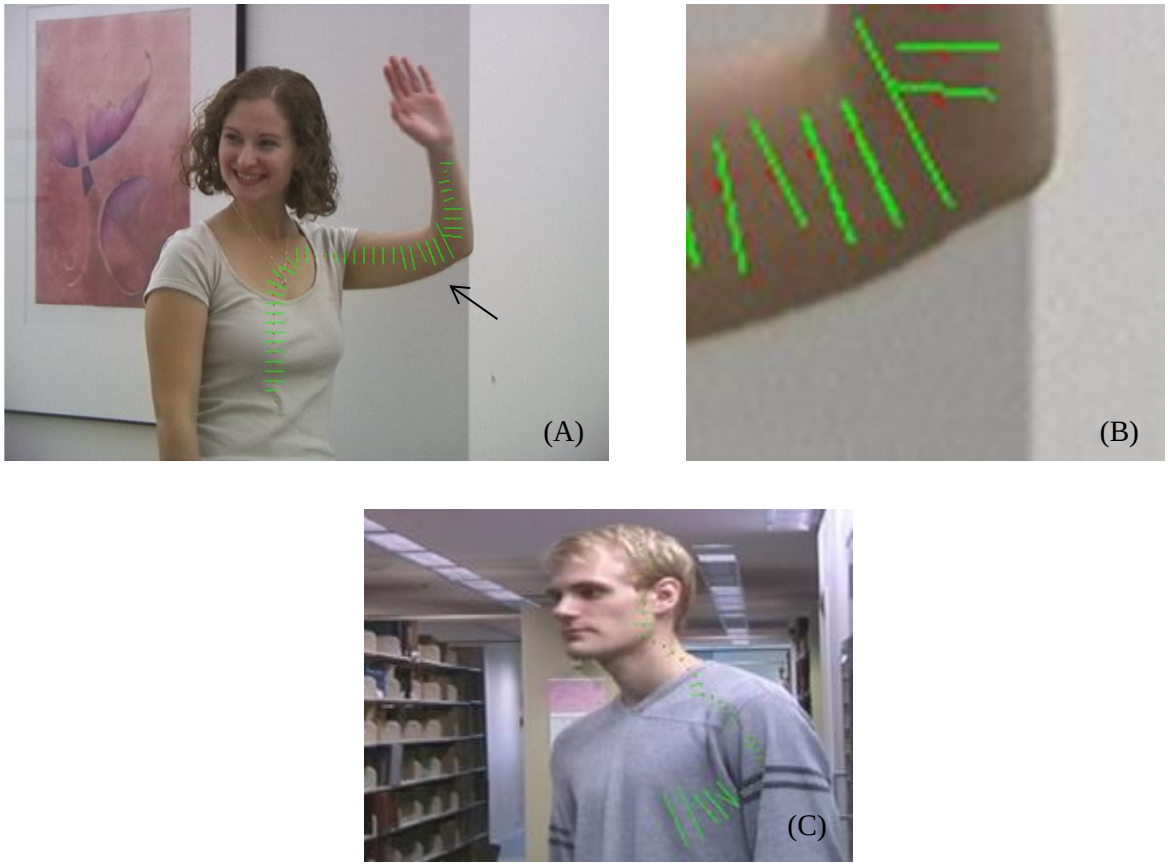


FIGURE 57 : (A)(C) LES SEGMENTS DE SIMILARITES DESSINES EN CHAQUE POINT DE LA COURBE.(B) ZOOM SUR (A)

4.5 MISE EN ŒUVRE DE LA PROPAGATION PAR COURBE ACTIVE

Le processus de propagation part de la courbe dessinée par l'utilisateur pour trouver la meilleure déformation permettant d'obtenir une courbe représentant au mieux l'objet initial. Successivement, le même processus est réitéré pour propager le marquage obtenu et ainsi obtenir un nouveau marquage sur la frame qui suit.

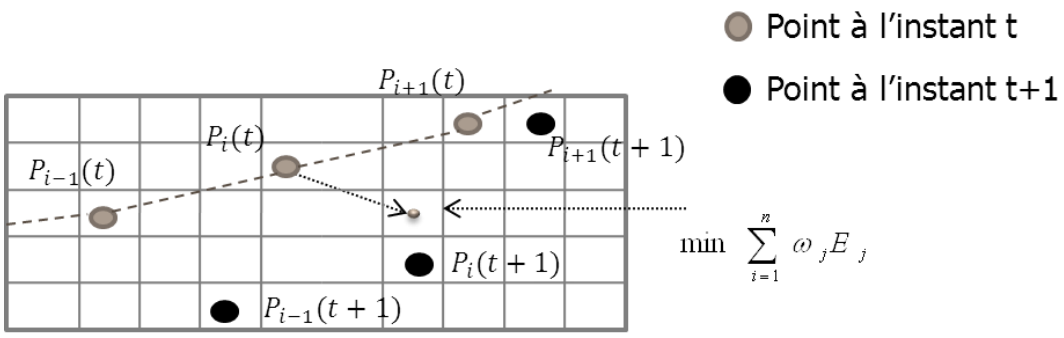


FIGURE 58 : PROCESSUS D'OPTIMISATION ESPACE TEMPS

Basée sur l'ensemble de contraintes que nous avons définies et représentées par un ensemble de cinq termes d'énergies : deux termes composant l'énergie interne et trois termes composant l'énergie externe, l'énergie globale dont les termes sont pondérés par différents poids ω_i que nous avons définis s'écrit comme suit :

$$E_{globale}(C) = \sum_{p \in C} \begin{pmatrix} \omega_1 E_{unif}(p) + \omega_2 E_{courb}(p) + \\ \omega_3 E_{mvt}(p) + \omega_4 E_{couleur}(p) + \omega_5 E_{stablité}(p) \end{pmatrix} \quad (30)$$

Les différents poids ont été déterminés expérimentalement, comme c'est le cas dans la plupart des algorithmes basés sur la minimisation d'énergie. Les résultats basés sur cette approche, en gardant le même ensemble de poids fixes, seront présentés dans le chapitre 5. L'utilisation de poids fixes et non évolutifs n'est pas la meilleure approche. Dans la section 4.5, nous verrons les mécanismes que nous avons mis en place et qui nous ont permis d'améliorer les performances de cette approche.

4.6 GESTION DYNAMIQUE DES POIDS

La définition des poids associés aux énergies de la fonctionnelle à minimiser est un des principaux problèmes connus dans la plupart des modélisations par courbe active. C'est un problème qui n'est pas souvent discuté. En effet, la majorité des approches se contente d'accomplir un grand nombre d'essais afin de fixer un ensemble de poids qui fera le juste équilibre pour que la modélisation soit fidèle à un type particulier d'images et de problématique. Dans la plupart des cas nous trouvons que les approches basées sur les courbes actives s'accompagnent soit d'une interface graphique soit d'un fichier où l'utilisateur final pourra redéfinir les poids pour coller au plus à son système. Ceci n'est pas la tâche d'un utilisateur complètement novice, de réaliser et d'affiner lui-même ces réglages. Dans plusieurs

études on remédie à cette problématique par la définition de différents ensembles de poids adaptés chacun à un type d'image. Une règle générale indiquant comment définir ces poids n'existe pas.

La difficulté de fixer les poids est issue de différentes causes. Avec deux ensembles de poids différents sur une même image donnée, nous pouvons aboutir au même résultat. De manière classique, les différents poids sont choisis en se basant sur une connaissance préalable ou une intuition du concepteur du système sur le processus, suivie d'une phase d'estimation empirique intervenant alors pour permettre la détermination des valeurs les plus appropriées au problème qu'on traite.

Des études plus récentes [Etyngier et al. 2007] se sont intéressées à ce problème et ont pu, en utilisant des connaissances *a priori*, définir un protocole d'apprentissage permettant de définir l'ensemble des poids de façon robuste pour qu'ils soient adaptés au problème traité. Une telle approche ne peut pas être utilisée dans notre cas. D'une part les utilisateurs attendent un outil fiable, simple d'utilisation et rapide. D'autre part, la variabilité des objets que nous traitons et l'aspect interactif de notre application ne permettent pas ce type de mise en œuvre basée sur l'apprentissage. De plus, dans le cas où nous traitons de la vidéo, et donc un contenu avec une dimension temporelle non contrôlée (le temps séparant deux acquisitions d'image varie d'une caméra à une autre), nous pouvons noter que même si une force est effectivement plus importante que les autres, cela peut ne pas être tout le temps vrai pour toutes les frames de la séquence. Cette définition rigide pourrait même induire des erreurs de plus en plus importantes au fur et à mesure de l'avancement dans la vidéo. Par ailleurs, la courbe traverse des régions variées et peut prendre des formes de courbures variables aussi les poids optimaux choisis et définis de manière globale peuvent ne pas correspondre à tous les points d'une même courbe. Dans les travaux de [Rousselle 2003] nous pouvons remarquer une démarche vers la dynamisation des poids par l'introduction des contours actifs autonomes. Les poids sont considérés comme la composition de deux termes : un fixe et un variable que nous pouvons intégrer dans le processus d'optimisation. Les auteurs ont redéfini la fonction d'énergie globale comme suit :

$$E(M_1, \ldots, M_N, v_{ij}) = \sum_{i=1}^{p} \sum_{j=1}^{N} v_{ij}(M_i) F_j(M_i) \tag{31}$$

Où $M_i$est un point de la courbe, P le nombre de points de la courbe, N le nombre de contraintes utilisées et F une fonction représentant une énergie donnée. Bien que cette approche semble très prometteuse, d'après les auteurs elle est très gourmande en terme de temps d'exécution. En effet l'optimisation ne se faisant plus sur un seul critère, qui était le point de la courbe mais maintenant

aussi sur les poids qui lui sont associés. Le temps d'exécution par rapport à l'approche classique se voit multiplié même pour des cas assez simples.

Tel que décrit précédemment, le marquage dessiné par l'utilisateur n'a aucune propriété générique. La courbe peut traverser différentes régions de différentes textures. Une paramétrisation globale n'est pas alors très avantageuse dans notre cas. Nous n'avons aucun a priori ni sur la courbe, ni sur l'objet que l'utilisateur veut désigner. Pour résoudre ce problème nous avons introduit un mécanisme de gestion de poids de manière dynamique qui est basé sur les règles suivantes :

- Si le point que nous sommes en train de traiter est situé dans une région qui contient un gradient élevé, ce point peut être considéré comme un point saillant au sens usuel et peut être facilement suivi en utilisant les méthodes d'estimation de mouvement classiques, alors la première énergie relative aux déplacements doit être privilégiée.

- Si ce n'est pas le cas, le point considéré est dans une zone homogène dans laquelle un estimateur de mouvement risque d'induire le système en erreur et il vaut mieux alors accorder plus d'importance aux autres termes d'attache aux données.

Selon la position du point, la balance des poids entre les forces externes (relatives aux données) doit tenir compte de la situation dans laquelle chaque point se trouve. Le problème est divisé en sous-parties et l'optimal global est, lorsque les poids sont fixes, une somme à pondération égale, quel que soit le point considéré, de meilleures solutions soit localement possibles. Nous voulons changer dynamiquement les poids ponctuellement afin de coller vraiment à la nature de la portion d'image que nous sommes en train de traiter.

Précédemment nous avons défini que l'énergie globale $E_{globale}$ peut s'écrire sous cette forme :

$$E_{globale}(C) = \sum_{p \in C} \sum_{i=1}^{n} \omega_i E_i(p) \tag{32}$$

Avec ω_i une valeur réelle constante, indépendante des données considérées (donc de l'image) et de manière générale fixée globalement selon la problématique. Pour introduire une gestion locale des poids autour de chaque point p de la courbe nous pouvons réécrire l'équation d'énergie globale de la manière suivante :

$$E_{globale}(C) = \sum_{p \in C} \sum_{i=1}^{n} f\omega_i(p) E_i(p) \tag{33}$$

D'où nous écrivons :

$$E_{Dynglobale}(C) =$$

$$\sum_{p \in C} \begin{pmatrix} f\omega_1(p)E_{unif}(p) + f\omega_2(p)E_{courb}(p) + \\ f\omega_3(p)E_{mvt}(p) + f\omega_4(p)E_{couleur}(p) + \\ f\omega_5(p)E_{stabilité}(p) \end{pmatrix} \quad (34)$$

Ainsi, chaque valeur de poids est maintenant dépendante du point p auquel la force correspondante s'applique et obtenue au travers de la fonction $f\omega_i$ correspondante. Il nous faut alors définir des fonctions permettant de caractériser chaque point individuellement selon sa position (x, y) et son environnement dans l'image. Les valeurs représentant les poids obtenus via ces fonctions ne dépendent pas uniquement de la position dans la courbe mais elle peut prendre aussi en considération la position du point en question dans l'image, sa couleur ou aussi prendre en compte une zone de voisinage autour de celui-ci.

Plusieurs techniques peuvent être mises en place pour calculer les valeurs des ω_i sur chaque point de la courbe. Selon l'ensemble des forces définies, certaines $f\omega_i$ peuvent être définies comme étant constantes. Dans la plupart des cas, on a tendance à utiliser des valeurs constantes pour les énergies associées aux forces internes. Ceci s'explique par le fait que ces forces sont définies de façon indépendante des données contenues dans l'image. Ce sont les forces extrinsèques qui assurent l'attache aux données et qui sont susceptibles d'être touchées par l'adaptation dynamique des poids relativement aux données.

Une manière de caractériser un point, et ainsi de privilégier une force qui serait plus appropriée que les autres, peut se faire par la considération du gradient des points le long de la courbe. Nous pouvons, ainsi, classer les points en points à fort gradient et d'autres progressivement de plus en plus faible. Mais pour prendre en compte le fait que la courbe, pour désigner un élément, doit être au milieu de celui-ci, cette caractérisation peut se faire à travers les perpendiculaires à la courbe. Pour ce faire, nous avons choisi d'utiliser les segments de similarités, S_i, décrits dans la section 4.4.4,(Figure 57). Nous pouvons dire, alors, que plus la longueur de S_i est petite, plus grande est la probabilité que le point p_i soit un point de contour (contour de zone ou d'un élément). Ainsi, nous accordons plus ou moins confiance à une énergie, en se basant sur ce critère, par exemple. Nous pouvons définir pour l'énergie associée au déplacement, E_{mvt}, plus d'importance si le point p est un point de contour. Cela se fait par l'augmentation du poids qui lui est associé, $f\omega_3$, proportionnellement à la longueur de S_i, dénotée

$Ls(p)$. Par ailleurs, dans le cas où $Ls(p)$ est grand, la force de stabilité et la similarité couleur sont plus logiques à promouvoir par l'augmentation des poids qui leur sont associés, ω_4 et ω_5. En pratique, toutes les longueurs de segment de similarité $Ls(p)$ sont normalisées par le maximum de ces longueurs sur toute la longueur de la courbe :

$$fw(p) = {Ls(p)}\Big/{\max(Ls(p))} \tag{35}$$

L'énergie globale associée à notre problème peut se réécrire alors par la formule suivante :

$$E_{Dynglobale}(C) =$$

$$\sum_{p\in C}\begin{pmatrix}\omega_1(p)E_{unif}(p) + \omega_2(p)E_{courb}(p) + \\ (1-NLs(p))\,\omega_3 E_{mvt}(p) + \\ NLs(p)\omega_4 E_{couleur}(p) + \\ NLs(p)\,\omega_5 E_{stabilité}(p)\end{pmatrix} \tag{36}$$

En ce qui concerne les poids intrinsèques, nous proposons d'étudier la distribution des courbures du marquage dessiné dans l'image initiale ou de manière plus intelligente nous pouvons faire en sorte que les poids utilisés pour une frame à l'instant *t* soient dépendants de ceux obtenus à la frame *t+1*. Cela n'est pas tout le temps une solution optimale car nous ne pouvons pas faire d'hypothèse sur la nature déformable des objets que nous traitons. Il est plus fiable de faire en sorte que la définition des poids intrinsèques soit faite de manière classique et d'utiliser le processus dynamique uniquement pour les énergies dépendantes des données de l'image.

4.7 CONCLUSION

Dans ce chapitre nous avons mis en place un mécanisme de propagation de courbe basé sur les courbes actives. Cette modélisation a été mise en place suite à une définition des contraintes associées à notre problématique. La première étape a été autour de la définition des contraintes et de leur transcription dans une formulation énergétique. Une telle formulation, une fois la minimisation réalisée, nous permet d'aboutir à une déformation du marquage initial pour obtenir la meilleure position qui le représente à l'instant t+1 et successivement, ainsi. Les modélisations classiques par courbes actives se caractérisent par l'utilisation de poids afin de raffiner l'algorithme, spécifier l'importance d'une énergie face aux autres. Pour améliorer nos résultats, nous avons adopté une démarche dynamique rendant ces poids proportionnels aux données point par point le long de la courbe. Cela a permis à notre propagation d'être plus robuste, comme nous l'avons présenté dans la

section précédente. Dans la section suivante nous allons aborder la dernière étape de l'architecture que nous proposons pour le matting d'objet vidéo.

CHAPITRE 5. RESULTATS ET DISCUSSIONS DE L'APPLICATION AU MATING VIDEO

Dans ce chapitre, nous présentons nos résultats de propagation de marquage fait par l'utilisateur en se basant sur les deux philosophies présentées dans les chapitres 3 et 4. Par ailleurs, nous verrons quelles sont les limitations auxquelles nous nous sommes confrontés pour intégrer notre travail dans un processus rapide de *matting* vidéo. Nous présentons aussi les démarches et les résultats que nous avons obtenus, en particulier les améliorations apportées à l'algorithme *Spectral Matting* pour surmonter ces difficultés.

5.1 INTRODUCTION

Dans le monde de la production télévisuelle et cinématographique les moyens mis en œuvre sont souvent énormes. De plus, le travail de post-production nécessaire pour aboutir à des résultats assez simplistes est vraiment considérable. Ces tâches sont, en plus, effectuées par de grandes équipes spécialisées. Proposer des outils facilitant la phase de post-production permettrait de moderniser ce marché, qui est déjà la cible de plusieurs éditeurs de logiciels depuis quelques années. Cela permet aussi de pouvoir étendre la cible pour séduire le marché grand public. Une des techniques et notions qui peuvent être considérées comme un pilier essentiel dans le domaine de l'édition vidéo est la notion d'extraction d'objet vidéo. Extraire un objet vidéo, permet, par exemple de le copier dans une autre vidéo ou, par exemple, de remplacer l'arrière-plan de la vidéo. Dans ce contexte, nous présentons une application destinée à l'utilisateur final, produisant un rendu visuel du résultat pour l'extraction d'objet vidéo, ce dernier doit être bien soigné. Le *matting* offre une très intéressante alternative à la segmentation classique comme décrit précédemment. La complexité essentielle, de l'extraction d'objet vidéo, à laquelle nous avons tenté de répondre dans les chapitres trois et quatre est constitué par la phase de propagation de marquages et de l'intégration de la connaissance fournie par l'utilisateur dans les vidéos. Cette étape permet de réduire les efforts nécessaires et facilite la création d'applications simples en offrant un haut degré d'utilisabilité. Une fois les connaissances de l'utilisateur propagées pour atteindre toute les images de la vidéo, le problème de *matting* vidéo peut être réduit à un problème plus simple, mais pas pour autant facile, qui est le problème du *matting* d'images fixes.

Dans ce chapitre, nous allons, dans une première partie, présenter les résultats que nous avons obtenus dans le cadre de la propagation de marquages rentrés par l'utilisateur pour la désignation d'objet vidéo que ce soit par l'utilisation de la transformée en distance couleur ou par l'approche basée sur les courbes actives. Dans un deuxième temps, nous allons présenter l'approche Spectral *matting* permettant l'extraction d'un objet vidéo en se basant sur un seul type de marquage indiquant l'objet. Cela est différent des approches classiques présentées en section 2.5 qui nécessitent des marquages indiquant l'objet et des marquages indiquant le fond. Nous allons présenter la théorie du spectral *matting* mais aussi les limitations que nous avons rencontrées lors de la mise en œuvre de notre système de *matting* et les solutions que nous avons tenté de lui apporter.

5.2 RESULTATS DE LA PROPAGATION

Notre système propose une interface utilisateur permettant de dessiner un ou plusieurs marquages (gribouillis) sur n'importe quelle image de la vidéo pour désigner un objet dans la scène (l'objet, peut

être mobile et déformable et l'arrière-plan peut être dynamique). L'environnement n'est pas du tout maitrisé (dans le cas contraire d'autres approches tel que [Robinault et al. 2009] peuvent être plus adaptées). Deux modes existent : un mode de propagation automatique et un mode au pas. Dans le mode automatique, les marquages sont propagés en temps réel aux images suivantes de la vidéo. Dans le mode au pas, l'utilisateur peut intervenir à tout moment pour arrêter la propagation ou pour pointer une région de l'objet qui n'était pas visible auparavant.

Il est difficile de mesurer qualitativement le succès d'une approche telle que la nôtre. Une mesure possible peut être basée sur le nombre d'images dans lesquelles l'objet initialement indiqué par l'utilisateur continue à être désigné par notre algorithme. Nos résultats sont mieux visibles sous forme vidéo mais nous nous limitons, dans cette section, à présenter quelques images. Nous avons testé nos approches sur trois séquences standard de la littérature [Curless et al. 2001; Bai et al. 2009](Figure 59, Figure 65 et Figure 62) (séquences de personnes qui marchent, filmées par une caméra mobile 'fond dynamique').

Nous allons commencer par la présentation de nos premiers résultats obtenus en utilisant la transformée en distance couleur et par la suite nous montrerons les améliorations obtenues grâce à l'emploi de notre approche basée sur le modèle des courbes actives.

5.2.1 PROPAGATION PAR CDT

L'application de la transformation en distance couleur nous a permis d'obtenir une empreinte de l'interaction de l'utilisateur à partir d'une frame d'une séquence vidéo au reste de la séquence. Le processus que l'on a décrit au long de ce chapitre nous a permis de passer d'une empreinte et donc des images obtenues par la CDT à une courbe par image simulant chacune une désignation de l'objet initial dans les images suivantes de la séquence vidéo.

Comme précédemment décrit dans la section 3.2.1 la carte de distance couleur nous permet d'extraire l'empreinte d'un objet qu'on ne connaît pas. Cette empreinte extraite se transforme à son tour en un ensemble de courbes ou de gribouillis servant à designer l'objet initialement désigné par l'utilisateur. La validation d'un tel processus peut se faire de manière visuelle et donc par un retour d'expérience de la part de la personne ayant fait l'interaction initiale. Une mesure plus quantitative par laquelle on peut procéder est de comptabiliser le nombre d'images dans une séquence donnée dans lesquelles la courbe désignant un objet continue à le désigner au fur et à mesure de la propagation. Dans la Figure 59 nous pouvons voir le résultat de la propagation d'un marquage fait par l'utilisateur sur la première frame et que ce marquage s'est propagé correctement jusqu'à la frame 15. Correctement signifie bien sûr qu'il

continue à désigner le même objet déformable désigné par l'utilisateur. Sur la Figure 60, nous pouvons voir que notre méthode nous a permis de propager le marquage initial jusqu'à la frame 18 tout en désignant l'objet initial. La différence de performance vient en effet de la nature du marqueur et des mouvements dans la scène. Le marquage fait par l'utilisateur dans la Figure 60 est plus simple du point du vue sa forme et les déplacements et les déformations que subit l'objet d'intérêt sont capturés par l'empreinte du marquage initial, ce qui fait que la propagation est meilleure. Nous pouvons bien voir que le marquage s'est raccourci au niveau du visage car la partie marquée n'est plus visible et qu'il a suivi la courbure du cou alors que pour l'exemple de la Figure 59, le marquage est plus complexe et n'arrive pas à se déformer assez.

FIGURE 59 : PROPAGATION PAR CDT D'UN MARQUAGE FAIT PAR L'UTILISATEUR A PARTIR DE LA PREMIERE IMAGE A LA FRAME 4, 11 ET 15

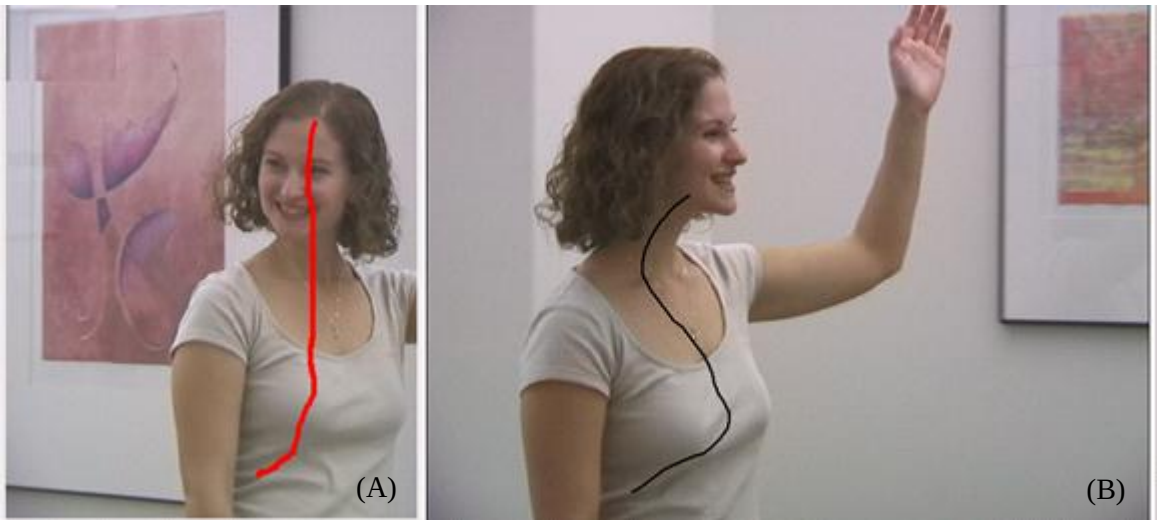


FIGURE 60 : PROPAGATION PAR CDT D'UN MARQUE FAIT PAR L'UTILISATEUR DE LA FRAME 1 A FRAME 18

Marquage en entrée	Nombre d'images où le marquage est correctement propagé par utilisation de la CDT
Marquage 1 (Figure 59-B)	15
Marquage 2 (Figure 60-A)	18

5.2.2 PROPAGATION PAR COURBES ACTIVES

DANS LE DOMAINE DU *MATTING* VIDEO, IL N'EXISTE PAS D'APPROCHES PROPAGEANT LES MARQUAGES FAITS L'UTILISATEUR. LA PHASE DE PROPAGATION EST SOUVENT PLUS COMPLEXE ET PLUS PROCHE DE LA DIFFERENCE ENTRE DEUX CLASSES OU SOUS-PARTIES DE L'IMAGE (AVANT-PLAN/ARRIERE-PLAN) QUE DE LA PROPAGATION DE SIMPLES MARQUAGES OU GRIBOUILLIS INDIQUANT L'OBJET. POUR COMPARER NOTRE AVONS IMPLEMENTE UN PROPAGATEUR DE GRIBOUILLIS BASE SUR LE FLUX OPTIQUE, QUE NOUS NOTONS OFBP. LES FIGURE 61-A, FIGURE 65-A ET FIGURE 62-A, MONTRENT DES EXEMPLES DE GRIBOUILLIS EN ENTREE QUE NOUS ALLONS PROPAGER POUR CONTINUER A DESIGNER L'OBJET INITIALEMENT DESIGNE PAR L'UTILISATEUR. NOS RESULTATS SONT PRESENTES ET COMPARES A L'IMPLEMENTATION OFBP DANS LE

Tableau 3. Par rapport à notre première approche nous pouvons noter une large amélioration dans les résultats mais aussi dans le fait que nous pouvons traiter plusieurs marquages en parallèle et simultanément. Cette nouvelle modélisation, peut être divisée en deux parties : modélisation par courbes actives et modélisation par courbes actives avec gestion de poids dynamiques. Nous avons

évalué que le mécanisme de la gestion dynamique des poids nous apporte nous apporte un gain de 20% dans la séquence Amira (Figure 61) et de 37% dans la séquence Walking Man (Figure 62). De manière générale, nous pouvons notre modélisation est efficace, La Figure 63 montre un exemple de difficulté de propagation que nous pouvons rencontrer si nous ne considérons pas un processus d'optimisation globale par courbe entière.

FIGURE 61: PROPAGATION D'UN MARQUAGE FAIT PAR L'UTLISATEUR DE LA FRAME 1 AUX FRAMES 8, 23 ET 30

FIGURE 62: (A) L’UTILISATEUR DESIGNE LA PERSONNE DANS LAVIDEO PAR DEUX MARQUAGES SUR LA PREMIERE IMAGE. FRAMES (B)(C)(D)(E) LA PROPAGATION DES MARQUAGES INITIAUX SUCCESSIVEMENT AUX FRAMES 7, 14, 25 ET 30. (F) ZOOM PERMETTANT DEVOIR LES ERREURS QUI SE SONT PRODUITES A LA FIN

TABLEAU 3 : NOMBRE D'IMAGES DANS LESQUELLES L'OBJET INITIALEMENT CHOISI, CONTINUE A ETRE DESIGNE

Vidéo / Méthode	Amira (30 frames)	Adam Lib (29 frames)	Walking man (30 frames)
OFBP	11	26	5
Notre approche sans la gestion de poids dynamique.	24	29	14
Notre approche avec la gestion de poids dynamique	30	29	25

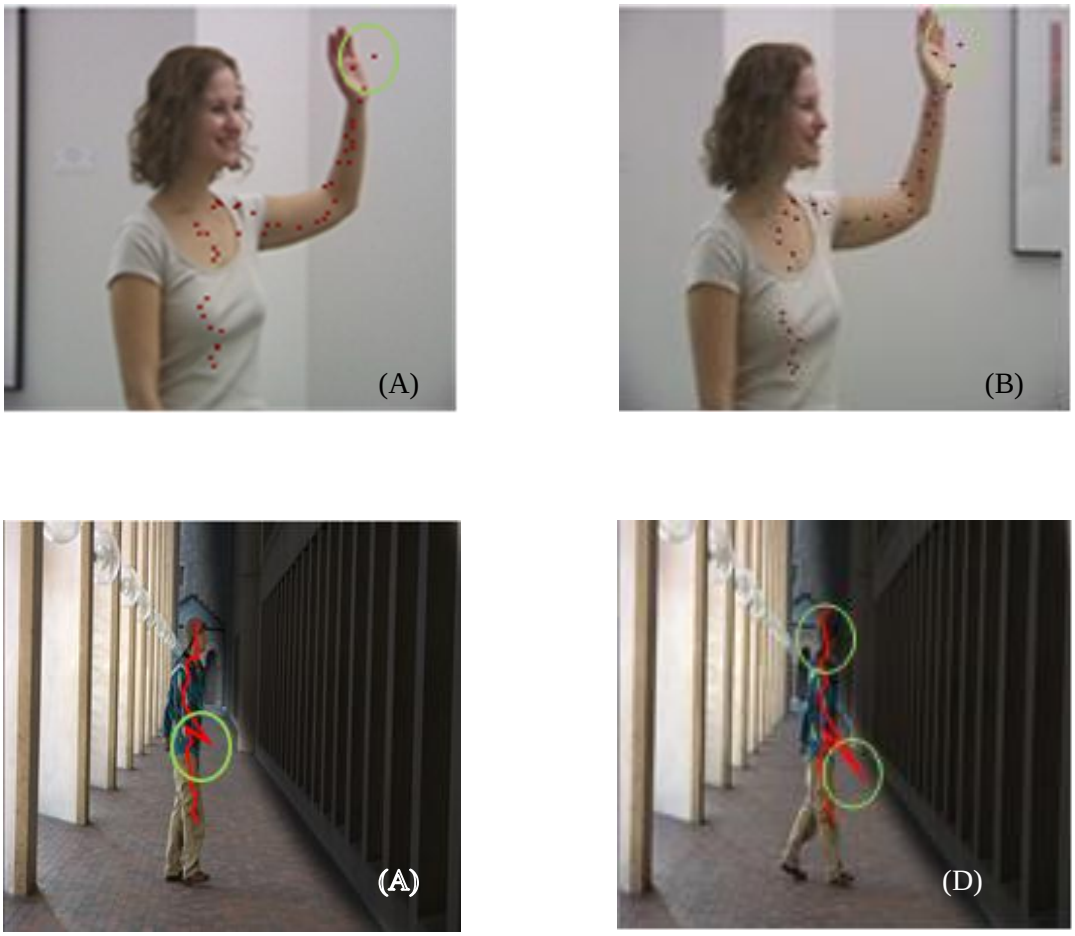


FIGURE 63: (A)(B) RESULTETS DE LA PROPAGATION BASEE SUR LE FLUX OPTIQUE LES ERREURS SONT VISUBLES A PARTIR DE LA FRAME 10 DE LA VIDEO AMIRA ET DE LA FRAME 5 DE LA VIDEO WALKING MAN. LES ERREURS SONT INDIQUEE PAR DES CERCLES VERTS.

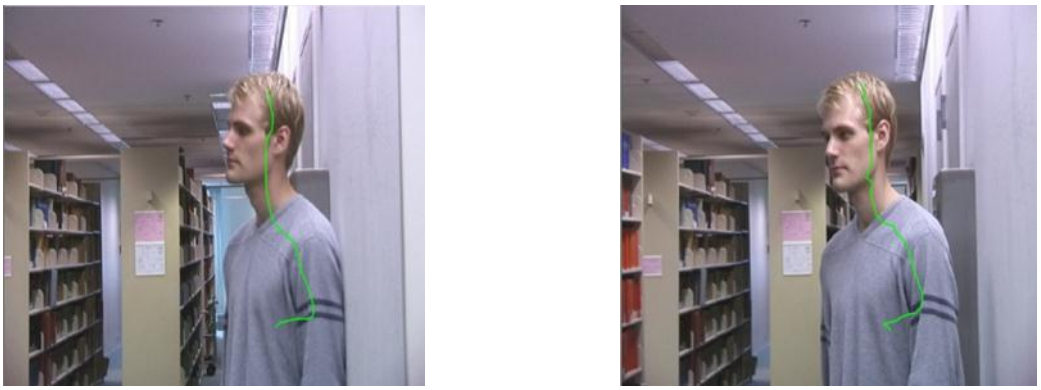


FIGURE 64 : PROPAGATION D'UN MARQUAGE FAIT PAR L'UTLISATEUR DE LA FRAME 1 AUX FRAMES 9, 19 ET 29

FIGURE 65: (A) L'UTILISATEUR DESSINE UN MARQUAGE EN PLUS POUR DESIGNER UNE PARTIE DE L'OBJET QUI N'ETAIT PAS VISIBLE AU DEBUT DE LA VIDEO. (B) LA PROPAGATION CONTINUE JUSQUÀ LA FRAME 29

.

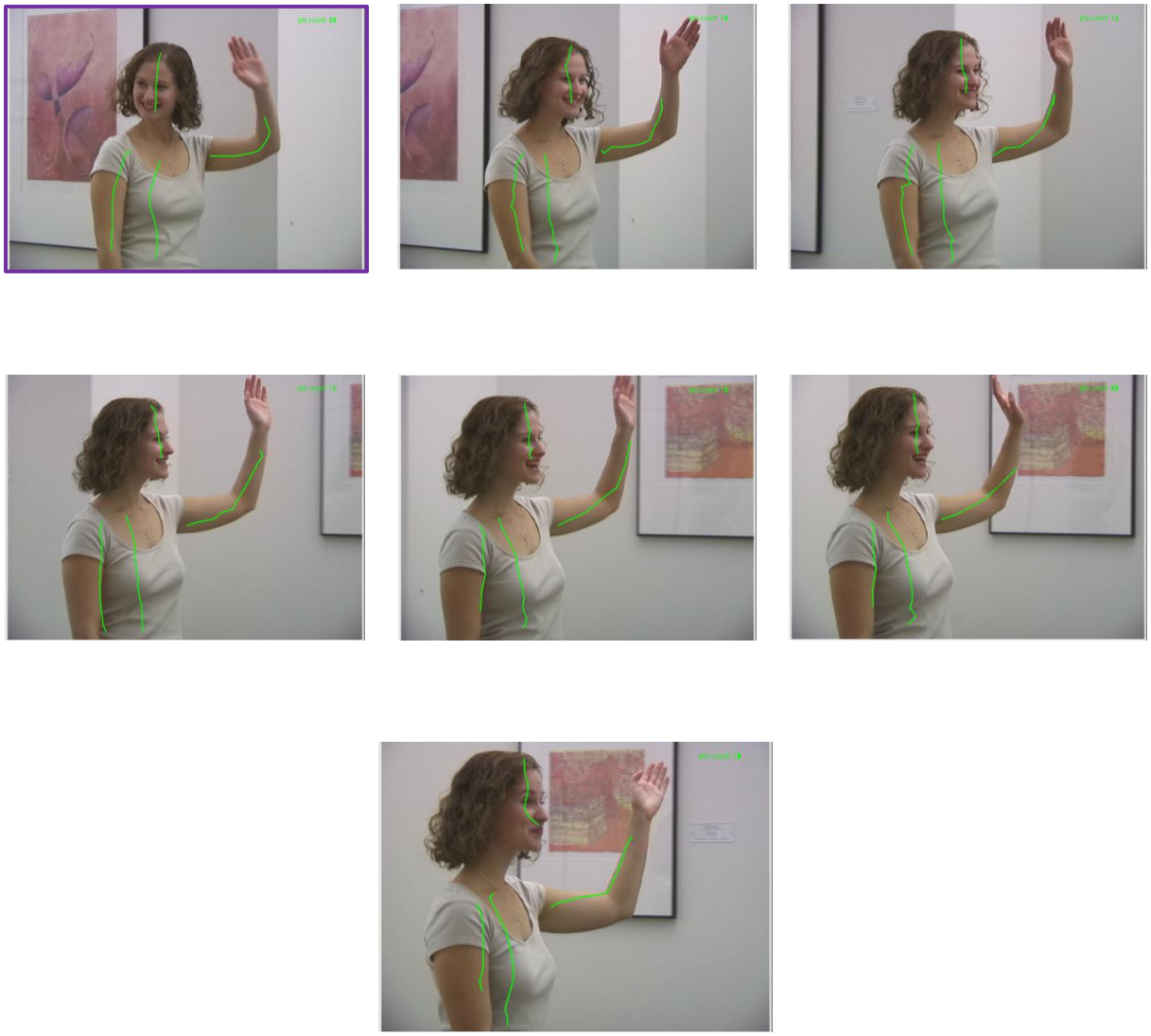


FIGURE 66 : PROPAGATION D'UN MARQUAGE FAIT PAR L'UTLISATEUR DE LA FRAME 1 JUSQU'A LA FRAMES 30

5.3 SPECTRAL MATTING : PRINCIPES ET LIMITATIONS

Dans la suite de ce chapitre nous allons aborder la problématique du spectral *matting*. Tel que décrit dans l'architecture que nous proposons pour la création d'un système d'extraction d'objet vidéo, le spectral *matting* prend comme entrée les marquages propagés par notre algorithme pour ensuite combiner l'ensemble de sur-segmentation qu'il génère et par la suite former l'objet. Dans cette section nous allons présenter les bases théoriques du spectral *matting* et nous allons ensuite mettre en avant

l'écart entre la théorie et la pratique en ce qui concerne les limitations de cette approche et les solutions que nous avons mises en place pour l'améliorer.

5.3.1 PRINCIPES

Le spectral *matting* constitue une évolution d'orientation dans le monde *matting*. En effet, l'image n'est plus considérée sous la forme d'une somme entre un objet et un arrière-plan pondérée respectivement par une valeur *alpha* et *1-alpha*. L'image est considérée en chaque point comme la combinaison d'éléments d'avant-plan. Dans cette section nous allons présenter les origines du spectral *matting* et son fonctionnement.

5.3.2 L'ORIGINE DU SPECTRAL MATTING

Le spectral *matting* est une technique basée sur l'hypothèse suivante : la couleur de chaque point d'une image est formée par une composition convexe des couleurs provenant d'éléments d'avant-plan formant celle-ci. Du point de vue technique, la formulation du spectral *matting* dérive des approches de classification spectrale, nous verrons dans la suite cette similarité.

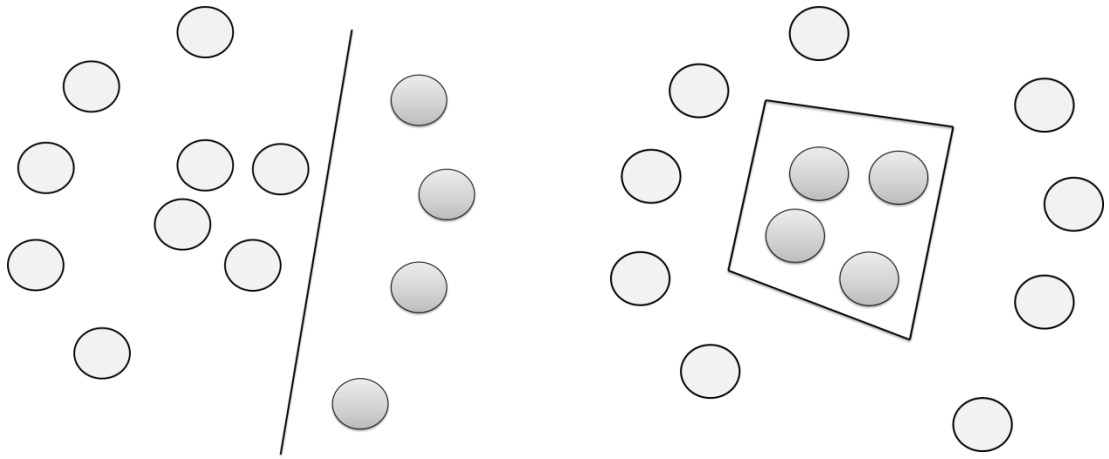


FIGURE 67 LA CLASSIFICATION SPECTRALE NE TIENT PAS COMPTE DE LA FORME, ELLE OPERE UNIQUEMENT SUR LE PRINCIPE DE PARTITIONNEMNT DE GRAPHE

La classification spectrale est une méthode de partitionnement non supervisé reposant sur la théorie des graphes et sur la minimisation de coupe. De manière générale, toute méthode de classification spectrale se décrit par les trois étapes suivantes [Verma and Meila 2003]:

- La création d'une matrice représentant la similarité des données.
- La détermination des vecteurs singuliers de la matrice de similarité.
- La définition des clusters en rapport aux vecteurs singuliers trouvés.

En premier lieu, dans les approches de segmentation spectrale, on commence par la transformation de l'image en un graphe pondéré non orienté. La matrice d'adjacence ou d'affinité A est obtenue par le calcul de a_{ij} le terme d'affinité entre les deux nœuds (pixels) voisins x_i et x_j en supposant que les nœuds sont numérotés de 1 à la taille de l'image en procédant ligne par ligne (le premier pixel de la deuxième ligne est le numéro w+1, avec w la largeur de l'image) :

$$a_{ij} = \begin{cases} W(i,j), & si\ i \neq j \\ 0, & sinon \end{cases} \tag{37}$$

La fonction W permet de mesurer la similarité entre deux nœuds du graphe et elle est représentée de manière générale par une gaussienne :

$$W(i,j) = \exp(-\frac{d(x_i, x_j)}{2\sigma^2}) \tag{38}$$

avec σ un paramètre de contrôle de W. il est à noter que plusieurs autres définitions de la mesure d'affinité sont possibles. Dans la littérature, il existe une multitude d'algorithmes définissant de manières différentes la matrice d'affinité, plus de détails peuvent être trouvés dans [Verma and Meila 2003][Malik 2000][Kannan et al. 2004].

Les méthodes de classification spectrale offrent un très bon potentiel pour la segmentation d'images. Ces approches ne permettent pas par contre la classification floue et ne sont pas satisfaisantes pour des applications dans le domaine du *matting*. Dans la section suivante nous allons voir comment A. Levin s'est inspirée de cette théorie de la classification spectrale pour l'étendre au *matting* d'images.

5.3.3 SPECTRAL MATTING

Le spectral *matting* propose une nouvelle formulation du problème du *matting* décrit dans la section 2.4.3, et propose une réécriture de l'équation (1) pour une image I, sous la forme suivante :

$$I = \sum_{k=1}^{K} \alpha_k F_k \tag{39}$$

Avec F_k un élément d'avant-plan et α_k sa transparence. Pour chaque point, la somme des α_k est égale à 1 ce qui garantit que l'image finale soit complète et cohérente. Levin et al. supposent que si on considère une région étroite de l'image, l'objet et le fond respectent une distinction de leurs composantes et chacune varie uniquement de point de vue de sa transparence. Dans [Levin et al. 2006], les auteurs ont introduit la notion de modèle de ligne couleur [Omer and Werman 2004] pour modéliser la problématique du *matting*. Cela suppose qu'au sein d'une petite région, w, centrée autour d'un pixel i, la couleur du fond et la couleur de l'objet sont uniformes et c'est la variation de la valeur alpha, la composante de transparence ou de mélange, qui génère la création de deux lignes de couleur dans l'espace RGB. Cela est formulé de la manière suivante :

$$\begin{cases} F_i = \beta_i^F F_i^1 + (1 - \beta_i^F) F_i^2 \\ B_i = \beta_i^B B_i^1 + (1 - \beta_i^B) B_i^2 \end{cases} \tag{40}$$

En partant de cette hypothèse, alpha peut être décrite par la combinaison linéaire convexe des valeurs de chaque sous-composant d'avant plan, I^c :

$$\sum\nolimits_c \alpha^c \mathrm{I}^c + b \,;\ \forall\, i \in w \tag{41}$$

Avec b une valeur constante au sein de la fenêtre. Ensuite pour extraire le masque alpha, les auteurs ont procédé à une étape d'optimisation de leur modèle défini ci-dessus de la manière suivante :

$$\mathrm{J}(\alpha) = \alpha^{\mathrm{t}} \mathrm{L}\, \alpha \tag{42}$$

où L est une matrice creuse semi-définie positive de taille NxN (avec le nombre de pixels de l'image) et L(i, j) est défini de la manière suivante :

$$A_q = \sum_{q|(i,j)\in w_q} \left(\delta_{ij} - \frac{1}{|w_q|} \left(1 + \left(I_j - \mu_q\right)^t \left(\Sigma_q + \frac{\varepsilon}{|w_q|} I_3 \right)^{-1} \left(I_j - \mu_q\right) \right) \right) \quad (43)$$

δ_{ij} représente le symbole de Kronecker, $\mu_q$est un vecteur représentant la moyenne des couleurs sur une fenêtre q, Σ_q est la covariance couleur et I_3 indique ici la matrice identité. Ainsi le L *matting* Laplacien s'écrit comme suit, comme la somme des relations d'affinités entre les pixels :

$$L = \sum_q A_q \quad (44)$$

Le problème du *matting* est ainsi réduit à l'optimisation de la fonction quadratique de coût, $J(\alpha) = \alpha^t L\, \alpha$. L'approche spectrale est utilisée pour analyser les vecteurs propres correspondant aux petites valeurs propres. Ensuite, une classification par *K-means* est utilisée afin d'effectuer une projection des éléments issus d'un vecteur propre donné dans l'espace propre lui correspondant, Figure 68.

FIGURE 68 : PASSAGE VIA CLASSIFICATION PAR TRANSFORMATION LINEAIRE DES VECTEURS PROPRES AUX '*MATTING COMPONANTS*'

Ainsi nous obtenons pour chaque image une décomposition en éléments d'avant-plan sous la forme de masques *alpha* formés de valeurs comprises entre 0 et 1, Figure 69.

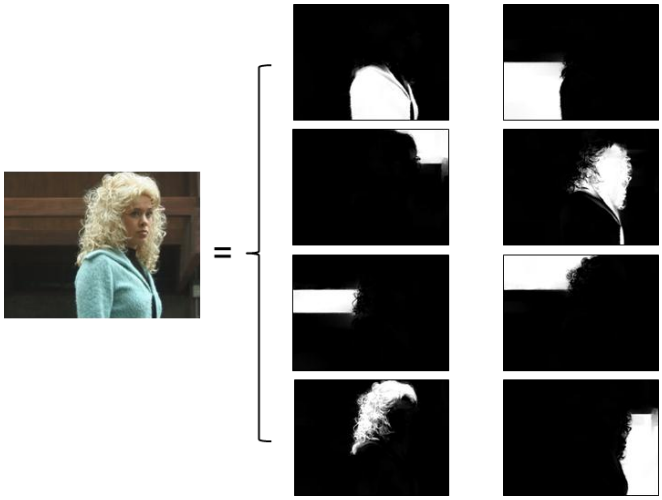


FIGURE 69: DECOMPOSITION EN MASQUES CONTINUS D'AVANT-PLAN

Le groupement de ces composants pour former le résultat final peut se faire par la simple composition d'éléments permettant de former l'objet d'intérêt. Cela est très rapidement faisable à travers une interface graphique permettant à l'utilisateur d'accomplir cette sélection. Les auteurs proposent, une approche alternative se basant sur l'optimisation des compositions, complétement automatique, pouvant fonctionner dans des cas simples, Figure 70.

FIGURE 70 : EXEMPLE DE RESULTAT OBTENU PAR L'APPLICATION DU SPECTRAL MATTING

5.3.4 LIMITATIONS

Un des problèmes auxquels de nombreuses approches de segmentation ou de *matting* sont confrontées vient de la ressemblance entre l'élément à extraire et le fond. Le spectral *matting* est aussi confronté à cette difficulté. De plus, comme dans le cas du spectral *matting*, on ne cherche pas à déterminer deux

classes d'éléments, mais plutôt un certain nombre de sous-éléments d'avant-plan (ce nombre est fixé manuellement, il n'est pas alors adapté à toutes les images) ce qui est encore plus compliqué. Un autre problème vient de l'hypothèse du modèle de ligne couleur qui n'est pas tout le temps respecté dans le cas des images naturelles. Les trois problèmes essentiels à surmonter sont donc :

- La vérification de la validité du modèle de ligne couleur face aux différents types ou causes de *matting* possible. Le *matting* peut venir, tel que décrit dans la section 2.4.3, de phénomènes physiques liés à l'acquisition mais aussi des effets de flou ou de flous de mouvement.

- La définition du nombre optimal de composantes pour aboutir à une bonne sur-segmentation de l'image. Une solution à ce problème a été proposée dans [Wu et al. 2012] par une classification itérative par *K-means* en décomposant l'image en sous-graphes.

- L'optimisation de la séparabilité des composantes et des pixels. Une solution intermédiaire pourrait être déjà l'optimisation de la séparation entre les classes et cela peut se faire à travers l'amélioration du calcul du terme d'affinité.

Pour l'application de notre processus de *matting* vidéo, décrit dans la section 1.3.3, l'étape finale consistant à la création de l'objet d'intérêt nécessite, la propagation du marquage fait par l'utilisateur et la décomposition ou la sur-segmentation de l'image en sous-composants d'avant-plan. Cette dernière étape basée sur l'algorithme spectral *matting* s'est avéré assez compliqué. En effet, nous pouvons remarquer sur la Figure 71, par exemple, que les résultats présentés par les auteurs nécessitent des conditions bien particulières qui ne sont pas expliquées par ces derniers. La deuxième ligne de la Figure 71 présente un résultat de *matting* présenté dans l'article semblant de bonne qualité. Sur la première ligne de la même figure nous pouvons voir le résultat que nous avons obtenu en utilisant la méthode proposée. L'extraction est complètement fausse. En effet les auteurs ont dû tronquer l'image d'origine pour aboutir à un *matting* correct de l'objet.

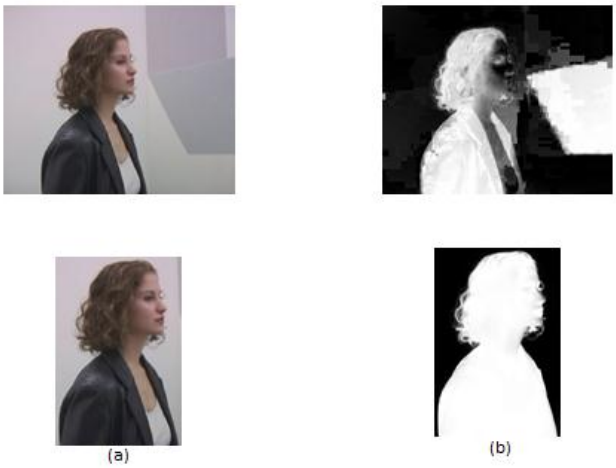


FIGURE 71 : PRIMERE LIGNE : APPLICATION DE LA METHODE [LEVIN ET AL. 2008] SUR UNE IMAGE DE LA SEQENCE AMIRA2 ET SON RESULTAT. LA DEXIEME LIGNE PRESENTE LA MEME APPLICATION EN TRONQUANT L'IMAGE

Nous nous sommes proposés d'essayer de résoudre les problématiques de [Levin et al. 2008] et ce qui semble pertinent, c'est que le Laplacien ne détecte pas correctement la séparation des clusters. Nous avons décidé d'ajouter des informations permettant de décrire mieux le manque d'homogénéité entre les pixels. Nous proposons d'intégrer les tenseurs couleur dans la formulation du spectral *matting* pour résoudre cette problématique.

5.4 AMELIORATION DU SPECTRAL MATTING BASEE SUR LES TENSEURS COULEUR

Une image couleur peut être considérée comme un champ de couleurs sur le plan x et y. Mathématiquement, le gradient d'une image peut être, alors, représenté par un tenseur. Ce dernier est un objet mathématique qui incarne la géométrie de l'espace image. Di Zenzo [Zenzo 1986] a introduit une définition du gradient d'une image multi-composante dans l'espace couleur RVB. Il propose une extension du gradient scalaire en utilisant des notations de tenseur. On passe d'un champ de scalaires à un champ vectoriel défini comme les variations locales de l'image vectorielle. Par rapport aux plans x et y, le vecteur gradient s'écrit de la manière suivante :

$$\nabla x = \left(\frac{I_R}{\partial x}, \frac{I_G}{\partial x}, \frac{I_B}{\partial x}\right)$$

$$\nabla y = \left(\frac{I_R}{\partial y}, \frac{I_G}{\partial y}, \frac{I_B}{\partial y}\right) \quad (45)$$

Pour trouver la direction des variations locales maximales et sa valeur associée, nous prenons la norme au carré de la variation vectorielle de l'image. Une image RGB peut être représentée par un tenseur. Le tenseur, T, s'écrit sous la forme matricielle suivante en utilisant la forme quadratique du gradient appelée première forme fondamentale [Bronshtein and Semendyayev 1997] d'une surface dans l'espace paramétrée par (x, y) :

$$f_x = \left|\frac{I_R}{\partial x}\right|^2 + \left|\frac{I_G}{\partial x}\right|^2 + \left|\frac{I_B}{\partial x}\right|^2$$

$$f_y = \left|\frac{I_R}{\partial y}\right|^2 + \left|\frac{I_G}{\partial y}\right|^2 + \left|\frac{I_B}{\partial y}\right|^2 \quad (46)$$

$$f_{xy} = \frac{I_R}{\partial x} \times \frac{I_R}{\partial y} + \frac{I_G}{\partial x} \times \frac{I_G}{\partial y} + \frac{I_B}{\partial x} \times \frac{I_B}{\partial y}$$

D'où :

$$T = \begin{bmatrix} f_x & f_{xy} \\ f_{xy} & f_y \end{bmatrix} \quad (47)$$

Après avoir défini la structure du tenseur couleur. Nous calculons les tenseurs de manière séquentielle pour chaque fenêtre 3x3, w_j. Cela nous permet d'extraire plus d'information sur l'affinité du voisinage. Pour une image scalaire, le tenseur T possède une seule valeur propre, λ_1, différente de zéro et qui est égale à la valeur maximale de la forme quadratique. Cette valeur est le carré du module du gradient, noté:

$$\|\nabla I\| = \sqrt{\lambda_1} \quad (48)$$

Cette information nous donne plus de connaissance sur la discontinuité entre les clusters de pixels et nous permet donc d'adapter ou d'améliorer l'estimation de l'affinité entre pixels voisins. Dans une fenêtre w_j, plus la valeur de $\|\nabla I\|$ est élevée, plus les pixels de celle-ci

auront tendance à être différents. Dans nos travaux, nous avons fixé de manière empirique un seuil, δ, qui nous a permis les meilleurs résultats possible sur la base d'images utilisées dans [Levin et al. 2008]. Nous avons défini que si l'amplitude du tenseur est supérieure à $\gamma = 0{,}09$, les pixels de la fenêtre en question ne proviennent pas d'un même cluster. Nous avons traduit cela dans la fonction d'affinité de la manière suivante :

$$A_q(i,j) = \begin{cases} \delta_{ij} - \frac{1}{w_q}\left(1 + (I_i - \mu_q)^t \times \|\nabla I\|_q{}^2\right)\left(\sum_q \frac{\varepsilon}{|w_q|} I_{3x3}\right)^{-1}\left((I_j - \mu_q) \times \|\nabla I\|_q{}^2\right); si\ \|\nabla I\| > 0.09 \\ \delta_{ij} - \frac{1}{w_q}\left(1 + (I_i - \mu_q)^t\right)\left(\sum_q \frac{\varepsilon}{|w_q|} I_{3x3}\right)^{-1}(I_j - \mu_q); si\ \|\nabla I\| < 0.09 \\ 0;\ sinon \end{cases} \quad (49)$$

La Figure 72 ci-dessous explique notre nouvelle démarche :

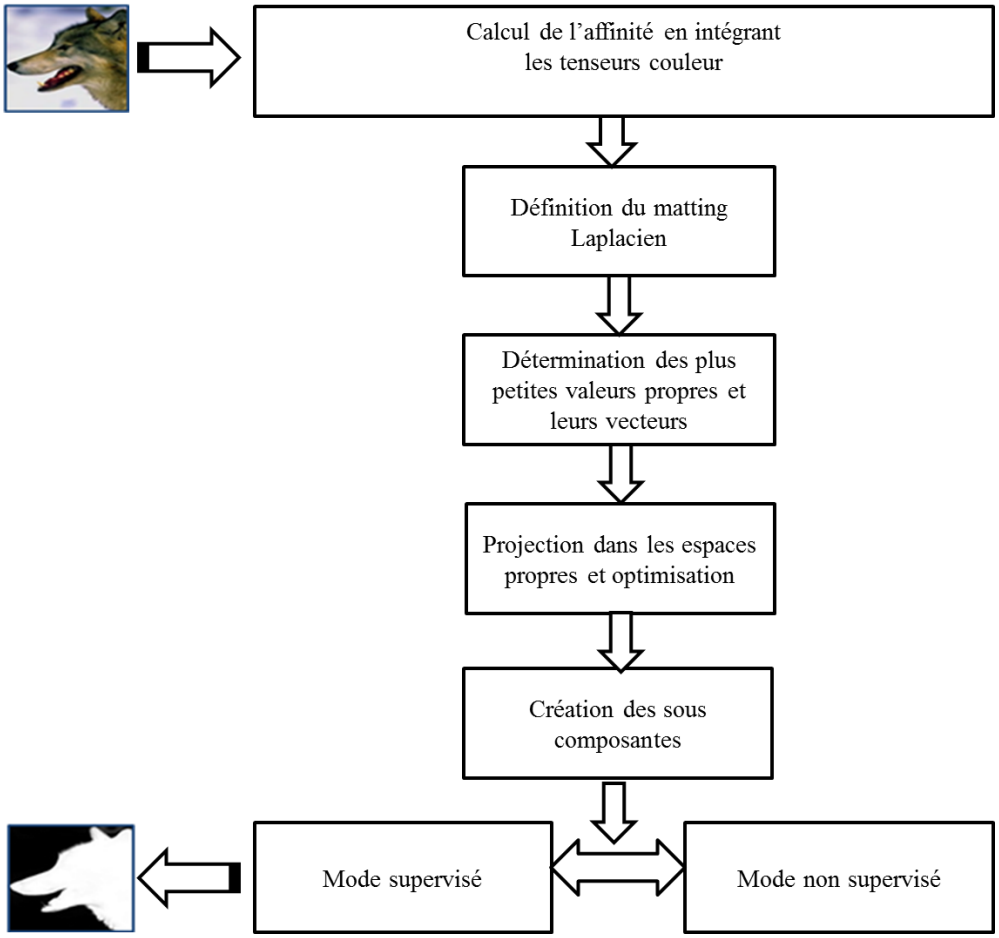


FIGURE 72 : DEMARCHE POUR L'AMELIORATION DU SPECTRAL MATTING

5.5 RESULTATS

Nous pouvons voir que les résultats que nous avons obtenus, Figure 73, par notre approche présentent des améliorations par rapport à ceux présentés par [Levin et al. 2008]. Cela dit, nous pouvons voir sur la **Erreur ! Source du renvoi introuvable.** que ces améliorations ne nous permettent pas d'atteindre e niveau de robustesse voulu et de pouvoir proposer un processus de *matting* vidéo complet basé sur cette approche.

FIGURE 73 : LA PREMIERE COLONNE CONTIENT LES IMAGES EN ENTREES, LA DEUXIEME COLONNE LES RESULTATS DE [LEVIN ET AL. 2008], LA TROISEME COLONNE LES RESULATATS OBTENUS PAR NOTRE APPROCHE ET LA DERNIERE LES VERITES TERRIN

FIGURE 74 : EXEMPLE D'IMAGE SUR LEQUEL NOTRE APPROCHE EST MOINS PERFORMANTE

CHAPITRE 6. CONCLUSION

Le secteur des outils centrés autour de l'utilisateur afin de lui fournir la facilité nécessaire permettant de promouvoir la consommation de données, de bande passante et ressources est en plein essor. Le monde de l'édition vidéo n'est pas épargné, au contraire il voit une très grande croissance pousser la popularisation des communications (*on-line* et *off-line*), des présentations, de l'éducation, des films, etc., sous le format vidéo.

L'objectif des travaux présentés dans cette thèse est d'apporter une première brique permettant de simplifier les outils de *matting* vidéo. Cela consiste en l'élaboration d'une approche permettant de simplifier la phase interactive nécessaire à ces outils en propageant les marquages faits par l'utilisateur pour la désignation d'objets vidéo.

Dans cette thèse nous nous sommes intéressés à la modélisation de l'interaction de l'utilisateur afin de proposer un mécanisme de propagation. Le marquage fait par l'utilisateur peut être vu comme une concentration des données composant l'objet. Nous avons présenté dans le chapitre 3 une méthode permettant de transformer le marquage, fait par l'utilisateur à un instant *t*, en un ensemble de données ou traces, possibles, couvrant un volume spatio-temporel. Ces traces permettent d'avoir des indications sur des régions dans lesquelles les marquages peuvent être localisés dans les images suivantes. Cela a été fait à travers une transformée en distance couleur que nous avons élaborés. A partir de ces informations propagées nous avons mis en place un processus qui nous a permis sur chaque image de reconcentrer et de condenser notre propagation pour aboutir de nouveau à un marquage désignant l'objet d'intérêt, comme si cela avait été dessiné par l'utilisateur. Cette approche était impactée par un manque de robustesse aux grandes déformations que l'objet peut subir en se déplaçant. Aussi, le fait d'accorder uniquement confiance aux informations chromatiques et à la localisation spatiale fait que cette approche n'est pas assez robuste face à la ressemblance de couleur entre l'objet et le fond. En partant de ces faits, nous avons décidé d'aborder le problème d'une manière différente en mieux modélisant les contraintes associées à la propagation de marquages, dans le cadre d'un objet mobile déformable et d'un arrière-plan dynamique. Nous avons mis en place une méthode

considérant le marquage comme une courbe formée par un ensemble de points reliés géométriquement pour former celle-ci. Nous avons considéré qu'une courbe représentant un marquage servant à l'utilisateur pour désigner un objet d'intérêt est soumise à différentes contraintes ou forces issue de notre modélisation, de notre problématique et du milieu dans lequel la courbe est plongée. Cela s'est traduit par une modélisation par courbe active qui, suite à un processus d'optimisation, nous permet d'obtenir successivement d'une image à la suivante un marquage représentant au mieux ce qu'aurait pu être le marquage dessiné initialement par l'utilisateur s'il avait dessiné sur l'image en cours.

Pour accomplir une extraction ou un *matting* d'objet vidéo, nous disposons par notre approche basée sur les courbes actives d'une base solide, légère et rapide pour la désignation semi-automatique d'objet vidéo. Dans la plupart des approches de *matting* vidéo actuelles (hormis quelques approches tel que [Weber et al. 2011; Bai et al. 2009]) l'automatisation des marquages se fait par interpolation et si ce n'est pas le cas, nous nous trouvons souvent face à des processus lourds qui propagent généralement l'objet d'intérêt dans sa globalité. Ce processus est suivi par une phase de *matting* qui permet d'obtenir une séparation parfaite entre le objet et le fond. Dans l'architecture que nous avons imaginée (décrite dans le chapitre 1) nous avons séparé le *matting* vidéo, aussi, en deux étapes : marquage et séparation de l'objet du fond. Nous avons par contre travaillé sur l'élaboration d'un processus simple et rapide qui ne propage pas un objet, mais uniquement les marquages le désignant. Cela suppose que nous avons besoin d'un seul type de marquage (marquage de l'avant-plan). Cette supposition implique que pour la phase de *matting* nous n'avons besoin que de ces contraintes-là qui nous décrivent l'objet et que nous n'avons pas besoin d'informations indiquant l'arrière-plan. L'approche proposée par A. Levin est la seule approche de *matting* d'images fixes satisfaisant ces conditions. Théoriquement cette approche constitue la brique finale nous permettant de proposer une approche de *matting* vidéo rapide et originale. Dans la pratique, l'application de cette approche sur les vidéos issues de l'état de l'art que nous avons utilisées ne donne pas les résultats attendus. Même lorsque nous avons utilisé la vidéo à partir de laquelle une image illustrant les résultats positifs de la méthode dans l'article[Levin et al. 2008], la méthode ne fonctionnait que sur cette image-là et la subtilité qu'on aperçoit pas à une première vue, c'est que cette image (la frame 12 de la séquence « Amira queen ») est utilisé tronquée des deux bord, sans cela, le résultat est mauvais (tel que présenté dans la Figure 71). Nous avons proposé des améliorations de la méthode [Levin et al. 2008], en améliorant le calcul du terme d'affinité (43) par l'intégration des tenseurs couleurs. Cela nous a permis d'améliorer la robustesse de la méthode sans pour autant arriver au stade où nous pouvons la combiner avec notre approche afin de proposer un système complet de *matting* vidéo.

Dans cette thèse nous avons travaillé sur la propagation de marquages dessinés par l'utilisateur pour la désignation d'objets d'intérêt. De plus nous avons présenté une amélioration de la méthode de *matting* d'image fixes[Levin et al. 2008]. Une piste de travail qui pourrait constituer une suite à ces travaux peut être dans l'intégration des deux approches de propagation que nous avons proposées. En effet, la transformée en distance couleur peut être intégrée pour la modélisation d'une force dans notre modélisation par courbes actives. D'autres applications peuvent, aussi, être étudiées, telles que l'application au suivi de geste. Un marquage peut être dessiné pour indiquer les jambes par exemple, ce qui nous permettra d'obtenir une sorte de squelette qui bouge. Cette application est actuellement limitée par le fait que nous ne gérons pas les occultations. Des améliorations dans ce sens sont aussi à prévoir

PUBLICATIONS DE L'AUTEUR

A. Ghorbel, M. Nouri, E. Marilly "Gradient color tensor based approach for spectral matting" 8th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP 2013), Barcelona, Spain, Feb 2013.

M. Nouri, O. Martinot, E. Marilly, N. Vincent "Dynamic weighting based active curve method for video object selection" 7th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP 2012), Rome, Italy, Feb 2012.

M. Nouri, E. Marilly, N. Vincent "Moving object selection based on an active curve approach" 18th IEEE International Conference on Image Processing (ICIP 2011), Bruxelles, Belgique, sept 2011.

M. Nouri, E. Marilly, N. Vincent "Sélection d'objet vidéo basé sur une approche active de propagation de gribouillis" 23ème GRETSI 2011, Bordeaux, France, sept 2011.

Brevets déposés:

N° de dépôt	**Titre**	**Couverture**
12305034.6	**User friendly video object extraction system with matting feedbacks** Inventeurs: Marwen Nouri et Gérard Delegue	**EP**
11171237.8	**Dynamic gesture recognition and authoring System** Inventeurs: Marwen Nouri, Emmanuel Marilly, Olivier Martinot et Nicole Vincent	**EP**

11305570.1	**Dynamic video quality adaptation for video analysis** Inventeurs: Gérard Delegue et Marwen Nouri	**EP**
12191465.9	**Method and device for allowing mobile communication equipments to access to multimedia streams played on multimedia screens** Inventeurs: Gérard Delegue et Marwen Nouri	**EP**
12191347.9	**Method for transmitting video content with high resolution to mobile communication equipements having collaborating screens and associated devices** Inventeurs: Saidi Mohamed Adel, Marwen Nouri, et Sayadi Bessem	**EP**
	Mono/multi resolutions adaptative video summarization and its adaptative delivery Inventeurs: Sayadi Bessem et Marwen Nouri	**EP**
12290095.4	**Method and equipement for achieving an automatic video summary of a video presentation** Inventeurs**:** Emmanuel Marilly, Oiliver Martinot et Marwen Nouri	**EP**
12305965.3	**A method, a server and a pointing device for enhancing presentations** Inventeurs: Marwen Nouri et Gérard Delegue	**EP**
12306332.3	**Age/Content adaptation checking system** Inventeurs: Marwen Nouri et Sayadi Bessem	**EP**

BIBLIOGRAPHIES

AMINI, A.., WEYMOUTH, T.. AND JAIN, T.., 1990. Using Dynamic Programming for Solving Variational Problems in Vision. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 12(9).

BAI, X. AND SAPIRO, GUILLERMO, 2007. A Geodesic Framework for Fast Interactive Image and Video Segmentation and Matting ∗. *IEEE ICCV*.

BAI, X., WANG, JUE, SIMONS, D. AND SAPIRO, G, 2009. Video snapcut: robust video object cutout using localized classifiers. *ACM Transactions on Graphics SIGRAPH*.

BAKER, S. AND MATTHEWS, I., 2004. Lucas-Kanade 20 years on: A unifying framework. part I: The quantity approximated, the warp update rule and the gradient descent approximation. *IEEE Computer Vision and Pattern Recongnetion, CVPR*, p.8.

BORGEFORS, G., 1986. Distance transformations in digital images. *Computer vision, graphics, and image processing*, 1(344), pp.344–371.

BOYKOV, Y.Y., 2001. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. *IEEE Computer Vision, ICCV*, 1, pp.105–112.

BRONSHTEIN, I.N. AND SEMENDYAYEV, K.A., 1997. *Handbook of Mathematics. 3rd ed* K. A. Kirs. Springer, ed.,

CHANG, S., 2007. Extracting skeletons from distance maps. *International Journal of Computer Science a*, 7(7).

CURLESS, B., SALESIN, D.H. AND SZELISKI, R., 2001. A Bayesian approach to digital matting. *IEEE Conference on Computer Vision and Pattern Recognition. CVPR*, 2, pp.II–264–II–271.

DAMASCHKE, P., 1995. The linear time recognition of digital arcs. *Pattern Recognition Letters*, 16, pp.543–548.

DANIELSSON, P.E., 1980. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14, pp.227–248.

EBERLY, D., 2001. Skeletonization of 2D Binary Images Characterization of Local Articulation Points. *Geometric Tools, LLC*, pp.1–11.

EKIN, A., TEKALP, A M. AND MEHROTRA, R., 2003. Automatic soccer video analysis and summarization. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 12(7), pp.796–807.

ELLIOTT, C., 2008. Bayesian Video Matting Using Motion Based Segmentation. *mcgill*, pp.1–16.

ETYNGIER, P., SEGONNE, F. AND KERIVEN, R., 2007. Active-contour-based image segmentation using machine learning techniques. *international conference on Medical image computing and computer-assisted intervention*, pp.891–899.

FOO, J.L., 2006. A survey of user interaction and automation in medical image segmentation methods. *Technical Report ISU-HCI-2006-2, Iowa State Universtity*.

FOULONNEAU, A., 2004. *Une contribution à l'introduction de contraintes géométriques dans les contours actifs orientés région*.

FOX, M. AND BERTSCH, G.F., 2002. *Optical properties of solids* Springer, ed.,

GASTAUD, M., 2005. *Modèles de contours actifs pour la segmentation d'images et de vidéos*. Université de Nice Sophia-Antipolis.

GRADY, L., SCHIWIETZ, T. AND AHARON, S., 2005. Random walks for interactive alpha-matting. *IASTED International Conference on Visualization, Imaging and Image Processing*, pp.423–429.

GUAN, Y., CHEN, W. AND LIANG, X., 2006. Easy Matting: A Stroke Based Approach for Continuous Image Matting. *Computer Graphics …*, 25(3), p.9600.

IRANI, M., 2002. Multi-frame correspondence estimation using subspace constraints. *International Journal of Computer Vision*, 48(153), pp.173–194.

JOSHI, N., MATUSIK, W. AND AVIDAN, S., 2006. Natural Video Matting using Camera Arrays. *ACM Transactions on Graphics*.

JUAN, O. AND KERIVEN, R., 2005. Trimap segmentation for fast and user-friendly alpha matting. *Variational, Geometric, and Level Set Methods in …*, (1), p.109.

KANNAN, R., VEMPALA, S. AND VETTA, A., 2004. On clusterings: Good, bad and spectral. *Journal of the ACM (JACM)*, 51(3), pp.497–515.

KASS, M., WITKIN, A. AND TERZOPOULOS, D., 1988. Snakes: Active Contour Models. *International Journal of Computer Vision*, pp.321–331.

LEFEVRE, S. AND VINCENT, N., 2004. Real time multiple object tracking based on active contours. *International Conference on Image Analysis and Recognition*, pp.606–613.

LEVIN, A., LISCHINSKI, D. AND WEISS, Y., 2006. A closed-form solution to natural image matting. *IEEE Computer Vision and Pattern Recongnetion, CVPR*, 1(2), pp.61–68.

LEVIN, A., RAV-ACHA, A. AND LISCHINSKI, D., 2008. Spectral matting. *IEEE transactions on pattern analysis and machine intelligence*, 30(10), pp.1699–712.

LI, Y., SUN, J. AND SHUM, H., 2005. Video object cut and paste. *ACM Transactions on Graphics (TOG)*, 2, pp.595–600.

LOUVAT, B., 2008. *Analyse de séquences d'images à cadence vidéo pour l'asservissement d'une caméra embarquée sur un drone*. Institut National Polytechnique de Grenoble.

MALIK, J., 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), pp.888–905.

MAO, K., ZHAO, P. AND TAN, P., 2006. Supervised learning-based cell image segmentation for p53 immunohistochemistry. *IEEE Biomedical Engineering*, 53(6), pp.1153–1163.

MARR, D., 1982. *Vision : a computational investigation into the human representation and processing of visual information* M. Press, ed.,

MELTER, R.A., 1991. *A survey of digital metrics* Springer, ed.,

OMER, I. AND WERMAN, M., 2004. Color lines: image specific color representation. *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2, pp.946–953.

OUTTAGARTS, A., SQUEDIN, S. AND MARTINOT, O., 2012. Keywords-based Automatic Multimedia Authoring in the Cloud. *ACM SIGMAP*.

PORTER, T. AND DUFF, T., 1984. Computer graphics. *ACM SIGGRAPH*, 18(3), pp.253–259.

POZ, A.P.D., VALE, G.M., PAULO, S., SIMONSEN, R.R. AND PRUDENTE-SP, P., 2003. *Dynamic programming approach for semi-automated road extraction from medium and hight resolution images* I. Archives, ed.,

PUDNEY, C., 1988. Distance-ordered homotopic thinning : A skeletonization algorithm for 3D digital images. *Computer vision and image understanding*, 72(3), pp.404–413.

QUWEIDER, M.K., SCARGLE, J.D. AND JACKSON, B., 2007. Grey level reduction for segmentation , threshholding and binarisation of images based on optimal partitioning on an interval. *IET*, 1(1), pp.103–111.

RAV-ACHA, A. AND PELEG, S., 2006. Lucas-Kanade without iterative warping. *IEEE ICIP*.

ROBINAULT, L., BRES, S. AND MIGUET, S., 2009. Real time foreground object detection using PTZ camera. *International Conference on Computer Vision, Theory and Applications*, pp.609–614.

ROTHER, C., KOLMOGOROV, V. AND BLAKE, A., 2004. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (TOG)*.

ROUSSELLE, J.J., 2003. *Les contours actifs: une méthode de segmentation*. Université Paris Descartes.

RUZON, M. AND TOMASI, C., 2000. Alpha estimation in natural images. *IEEE Computer Vision and Pattern Recongnetion, CVPR*, 1(June).

SAHOO, P.., SOLTANI, S. AND WONG, A. K.., 1988. A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, 41(2), pp.233–260.

SCHOLZ, V., EL-ABED, S., MAGNOR, M. AND SEIDEL, H., 2009. Editing Object Behavior in Video Sequences. *cgcf*, 0(0), pp.1–11.

SHI, J. AND TOMASI, C., 1994. Good features to track. *IEEE Conference on Computer Vision and Pattern Recognition*, pp.593–600.

SIMARD, P.-Y., LECUN, Y.A. AND DENKER, 1992. Efficient pattern recognition using a new transformation distance. *Advances in Neural Informatic Processing Systems*, pp.50–58.

SRAMEK, M. AND KAUFMAN, A., 2000. Fast ray-tracing of rectilinear volume data using distance transforms. *IEEE Visualization and Computer Graphics*, 6(3), pp.236–252.

VERMA, D. AND MEILA, M., 2003. A comparison of spectral clustering algorithms.

VINETTE, C., 2003. *L'information visuelle efficace pour la reconnaissance de visages dans l'espace-temps*. Université de Montréal.

VIOLA, P. AND JONES, M., 2001. Rapid object detection using a boosted cascade of simple features. *IEEE Computer Vision and Pattern Recongnetion, CVPR*, 1, pp.511–518.

WANG, J. AND COHEN, M.F., 2005. An iterative optimization approach for unified image segmentation and matting. *IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, (1), pp.936–943 Vol. 2.

WANG, JUE, BHAT, P., COLBURN, A. AND AGTAWALA, M., 2005. Interactive Video Cutout. *ACM SIGGRAPH*.

WANG, JUE AND COHEN, MICHAEL F, 2007a. Image and Video Matting : A Survey. *Computer Graphics and Vision*, pp.1–78.

WANG, JUE AND COHEN, MICHAEL F., 2007b. Optimized Color Sampling for Robust Matting. *IEEE Conference on Computer Vision and Pattern Recognition*, pp.1–8.

WEBER, J., LEFEVRE, S. AND GANCARSKI, P., 2011. Spatio-temporal quasi-flat zones for morphological video segmentation. *International Symposium on Mathematical Morphology*.

WILLIAMS, D. AND SHAH, M., 1992. A Fast Algorithm for Active Contours and Curvature Estimation. *CVGIP*, 55(1), pp.14–26.

WU, T., JUAN, H. AND LU, H.H., 2012. Improved spectral matting by iterative k-means clustering and the modularity measure. , (5), pp.1165–1168.

XU, C. AND PRINCE, J.L., 1998. Snakes, shapes, and gradient vector flow. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 7(3), pp.359–69.

YE, Q.Z., 1988. The Signed Euclidean Distance Transform and I t s Applications Qin-Zhong Ye Dept. of Electrical Engineering Linkiiping University. *IEE ICPR*, pp.495–499.

ZANCANARO, M., ROCCHI, C. AND STOCK, O., 2003. Automatic video composition. *Lecture notes in computer science*, pp.192–201.

ZENZO, S. DI, 1986. A note on the gradient of a multi-image. *Computer Vision, Graphics, and Image Processing*, 00, pp.116–125.

ZIOU, D. AND TABBONE, S., 1998. Edge detection techniques-an overview. *International Journal of Pattern Recognition and Image Analysis*, pp.1–41.