

Habilitation à Diriger les Recherches

Pour obtenir le diplôme d'Habilitation à Diriger les Recherches

Spécialité INFORMATIQUE

Préparée au sein de l'Université de Caen Normandie

Champs tensoriels pour la modélisation géométrique locale et non-locale des images et leurs applications

Présentée et soutenue par
David TSCHUMPERLÉ

Habilitation soutenue publiquement le 03/10/2018
devant le jury composé de :

Mr. Pierre KORNPORST	Chargé de recherche, INRIA Sophia-Antipolis	Rapporteur
Mr. Simon MASNOU	Professeur, Université Lyon I	Rapporteur
Mr. Patrick PEREZ	Directeur de recherche, INRIA Rennes - Directeur scientifique, VALEO.AI	Rapporteur
Mr. Rachid DERICHE	Directeur de recherche, INRIA Sophia-Antipolis	Examineur
Mme Christine FERNANDEZ-MALOIGNE	Professeur, Université de Poitiers	Examinatrice
Mr. Olivier LÉZORAY	Professeur, Université de Caen Normandie	Garant

Habilitation dirigée par Olivier LÉZORAY, laboratoire GREYC (UMR CNRS 6072), Caen



TABLE DES MATIÈRES

1	Images et tenseurs	1
1.1	Définition d'une image	1
1.2	Dérivées premières et secondes directionnelles	2
1.3	Les tenseurs du second ordre	3
2	Lissage anisotrope d'images guidé par champ de tenseurs	7
2.1	Introduction au lissage d'images	7
2.1.1	Historique des EDP pour le lissage d'images	8
2.1.2	Principe de fonctionnement	9
2.2	Géométrie locale des images multi-canaux	10
2.2.1	Quelles caractéristiques géométriques ?	10
2.2.2	Géométrie estimée à partir d'une caractéristique scalaire	11
2.2.3	Géométrie multi-valuée de Di Zeno	11
2.3	Lissage d'images multi-canaux par EDP	15
2.3.1	Méthodes variationnelles	15
2.3.2	EDP de type <i>Divergence</i>	17
2.3.3	EDP de type <i>Laplaciens Orientés</i>	19
2.3.4	EDP de type <i>Trace</i>	21
2.4	Conclusion du chapitre	24
3	EDP de diffusion et préservation des courbures	27
3.1	Régularisation à préservation de courbure	27
3.1.1	Cas d'une direction de lissage unique	28
3.1.2	Analogie avec les Convolutions de Lignes Intégrales	29
3.1.3	Une formulation entre <i>Traces</i> et <i>Divergences</i>	31
3.1.4	Extension au lissage guidé tenseurs (multi-directionnel)	32
3.2	Un schéma numérique d'implémentation efficace	33
3.3	Applications et résultats	36
3.4	Conclusion du chapitre	37

4	Un modèle tensoriel pour le rendu d'images en crayonnés	40
4.1	Transformations basées <i>traits</i>	40
4.2	Un algorithme de rendu crayonné d'images	42
4.2.1	Une première approche naïve pour le crayonné	42
4.2.2	Modélisation tensorielle de la géométrie des traits	43
4.2.3	Lancer de traits guidés par des tenseurs	44
4.3	Colorisation et extension artistique de l'algorithme	47
4.4	Conclusion du chapitre	53
5	Tenseurs pour l'inpainting d'images basé motifs	54
5.1	L'inpainting d'images basé motifs	54
5.2	Quelques contributions à l'état de l'art	61
5.2.1	Terme de données basé tenseur	61
5.2.2	Une stratégie de recherche astucieuse	63
5.2.3	Mélange spatial de patches	69
5.3	Résultats sur des images couleurs	73
5.4	Inpaining de paires stéréoscopiques avec carte de profondeur	74
5.4.1	Revue de l'état de l'art	78
5.4.2	Extensions proposées pour l'inpaining avec carte de profondeur	80
5.4.3	Résultats de complétion de vues virtuelles	83
6	Régularisation non-locale utilisant des tenseurs en grande dimension	87
6.1	Combiner EDP et méthodes basées patches ?	87
6.2	Une définition d'un espace de patches continu	89
6.3	Débruitage d'images par régularisation de Tikhonov basée patch	90
6.4	Analyse de la géométrie basée patch, dans Γ	93
6.5	Diffusion anisotrope dans l'espace des patches	94
6.6	Application et résultats	95
6.7	Conclusion du chapitre	96
7	Estimation de caractéristiques avancées en IRM de diffusion	99
7.1	Imagerie par résonance magnétique de diffusion	100
7.2	Reconstruction continue du signal de diffusion	102
7.3	Formalisme pour l'estimation de caractéristiques de la PDF	103
7.4	Application et résultats	105

7.5	Conclusion du chapitre	107
8	Développements logiciels pour le traitement d'images	108
8.1	CImg : une bibliothèque C++ générique pour le traitement d'images	109
8.1.1	Problématique, méthodologie et solutions proposées	109
8.1.2	Impact et statistiques d'utilisations	110
8.2	G'MIC : Un cadriciel multi-interfaces pour le traitement d'images	111
8.2.1	Problématique, méthodologie et solution proposée	111
8.2.2	Impact et statistiques d'utilisations	113
8.3	Expériences utilisateurs	116
9	Bilan et perspectives	124
9.1	Autres travaux	124
9.2	Projet de recherche	128
9.2.1	Contexte	128
9.2.2	Problématique	129
9.2.3	Objectifs	130
	Bibliographie	134
A	Curriculum Vitae	154
A.1	Vue d'ensemble de mon parcours de chercheur	157
A.2	Enseignement, formation et diffusion de la culture scientifique	159
A.3	Transfert technologique, relations industrielles et valorisation	162
A.4	Encadrement, animation et management de la recherche	163
B	Liste de mes publications, présentations et logiciels scientifiques	165
B.1	Livres et ouvrages (2)	165
B.2	Chapitres d'ouvrage (6)	165
B.3	Revue internationale à comité de lecture (10)	166
B.4	Actes de colloques internationaux à comité de lecture (32)	167
B.5	Actes de colloques nationaux à comité de lecture (19)	170
B.6	Logiciels (4)	172
B.7	Conférences invitées dans des congrès (18)	173
B.8	Rapports de Recherche (7)	175

Remerciements

Après avoir fait l'expérience de quatorze années de recherche intensive en traitement d'images au laboratoire GREYC de Caen, il m'est apparu que la liste des personnes que je devais remercier nécessiterait au moins un ou deux chapitres entiers dans ce manuscrit.

Etant quelqu'un de pragmatique, j'expérimente donc ici une technique relativement simple et efficace, qui va m'éviter ce travail de recensement colossal.

Un grand merci à tous !

Et tout particulièrement, à tous les membres du jury, qui ont accepté de prendre part à cette habilitation.

Comment est rédigé ce document ?

Ce manuscrit d'*Habilitation à Diriger les Recherches* compile un ensemble de travaux de recherche que j'ai mené depuis 2004, année de mon arrivée en tant que chargé de recherche CNRS dans l'équipe IMAGE du laboratoire GREYC de Caen.

La liste de ces travaux, effectués seul ou en tant qu'encadrant principal de doctorants/post-doctorants, n'est pas exhaustive : durant quatorze années, j'ai eu l'occasion d'aborder plusieurs sujets de recherche différents (et passionnants), mais parfois sans liens théoriques forts les uns avec les autres, hormis le fait qu'il s'agissait toujours de traitement d'images !

Pour l'écriture d'un manuscrit cohérent, j'ai donc procédé à une sélection de travaux qui mettent en lumière l'utilisation d'un modèle mathématique précis - à savoir les tenseurs du second ordre - pour la modélisation géométrique locale ou non-locale des images. Ainsi, chaque chapitre de ce manuscrit présente à chaque fois une problématique de traitement d'images différente, et des propositions de résolution de ces problèmes (ou d'améliorations de solutions existantes) qui se basent majoritairement sur l'utilisation de ces champs tensoriels.

Ces chapitres ne sont pas ordonnés chronologiquement, mais le sont par ordre de complexité croissante des modèles et des algorithmes étudiés, pour traiter des sujets de recherche exposés.

- Le chapitre 1, "**Images et tenseurs**", définit la modélisation mathématique utilisée pour les images et les tenseurs, ainsi que les calculs associés les plus fréquents, qui seront utilisés tout au long de ce manuscrit. On ne saurait en effet inclure dans un unique manuscrit tous les travaux existants mêlant tenseurs et images, et on se limitera donc seulement à certaines propriétés mathématiques de ces objets.
- Le chapitre 2, "**Lissage anisotrope d'image guidé par champs de tenseurs**", établit un état de l'art des méthodes de la littérature pour le lissage ou la régularisation d'images par équations aux dérivées partielles (EDP) de diffusion anisotrope. Ce rapide tour d'horizon fait naturellement le lien entre mes travaux de thèse et les travaux de recherche que j'ai pu mener par la suite, qui ont été grandement inspirés par les concepts (géométriques) développés dans ces formalismes de lissage d'images.
- Le chapitre 3, "**EDP de diffusion à préservation de courbures**", propose justement plusieurs améliorations significatives des approches classiques de lissage anisotrope guidé par champs de tenseurs : d'une part, pour traiter efficacement les configurations des structures à fortes courbures (coins, jonctions en T, . . .), et d'autre part, pour accélérer drastiquement le calcul de l'image résultante, via un schéma numérique d'implémentation astucieux.
- Le chapitre 4, "**Un modèle tensoriel pour le rendu d'image sous forme de crayonné**", utilise des modèles géométriques tensoriels très similaires à ceux décrits dans les chapitres 2 et 3, pour la génération de rendus non-photoréaliste d'images sous forme de dessins crayonnés. J'ai eu l'occasion de collaborer avec un artiste photographe, qui s'est approprié l'algorithme proposé pour réaliser des œuvres artistiques originales.
- Le chapitre 5, "**Des tenseurs pour l'inpainting d'image basé motifs**", apporte l'utilisation de champs tensoriels pour l'amélioration de divers points faibles des algorithmes d'inpainting d'images (*c-à-d.* la reconstruction de données manquantes) basés motifs. Ce travail m'a permis de co-encadrer (à 80%) simultanément mon deuxième doctorant (*Maxime*

Daisy) et un post-doctorant (*Pierre Buysens*), dans le cadre d'un projet "grand emprunt" 3DS d'envergure (9 partenaires académiques et industriels) portant sur les technologies du cinéma 3D, entre 2013 et 2015.

- Le chapitre 6, "**Régularisation non-locale d'images utilisant des tenseurs en grande dimension**", concerne l'utilisation de champs tensoriels pour la représentation de la géométrie *patchiale* d'une image, *c-à-d.* la géométrie de la surface formée par l'ensemble des patches de cette image, plongée dans un espace de patches spécifique Γ . Cet espace étant naturellement de *grande dimension*, j'ai dû utiliser des tenseurs étendus, à la fois pour l'analyse de la structure non-locale de l'image et pour la définition de comportements de diffusion anisotropes dans Γ (processus de diffusion donc non-locale).
- Le chapitre 7, "**Estimation de caractéristiques avancées en IRM de diffusion**", montre les limites du modèle tensoriel classique (pourtant vanté dans le reste de ce document !) pour la modalité d'IRM de diffusion qui permet d'analyser le mouvement des molécules d'eau dans les tissus (e.g. la matière blanche du cerveau). Nous avons dû formaliser un modèle géométrique plus complexe qu'un tenseur pour bien prendre en compte toutes les données acquises dans l'espace q (gradients de champ magnétique). Ce modèle basé sur des harmoniques sphériques et des polynômes de Laguerre, permet après adéquation du modèle aux données, d'estimer des caractéristiques quelconques de la *PDF* (densité de probabilité des déplacements) en chaque voxel de l'image. Ce travail correspond à mon premier co-encadrement (à 80%) de doctorant (*Haz-Edine Assemblal*), entre 2007 et 2010.
- Le chapitre 8, "**Développements logiciels pour le traitement d'image**", est un chapitre transverse, puisqu'il décrit les efforts de développement informatique que j'ai fourni au fil des années, entre autre pour réaliser les travaux présentés dans les chapitres précédents. J'ai toujours eu à cœur en effet de diffuser le résultat de mes travaux sous forme d'algorithmes reproductibles, libres à utiliser et à étudier. Cette volonté de partage m'a amené à développer petit à petit une bibliothèque C++ (*CImg*), ainsi qu'un logiciel multi-interfaces (*G'MIC*) pour le traitement des images, qui ont connus un certain succès, et qui comptent aujourd'hui plusieurs milliers de téléchargements et d'utilisateurs quotidiens.
- Le chapitre 9, "**Conclusions et perspectives**", résume mes contributions et définit mon projet de recherche, portant sur l'algorithmique du traitement d'images dans le *domaine artistique* (cinéma, photographie, illustration, peinture numérique, ...) et cherche à développer des méthodes *réseaux de neurones convolutionnels / méthodes géométriques* combinant les avantages des deux approches (qualité des résultats et empreinte mémoire réduite). Plusieurs collaborations sont envisagées avec des artistes de la communauté d'utilisateurs de mes logiciels, sur des sujets ouverts (colorisation automatique ou semi-automatique, inpainting, transfert de style ou de couleurs, ...) permettant de possibles transferts technologiques dans les plateformes logicielles dont je suis l'auteur.

Chaque chapitre débute par un résumé rapide de son contenu, la période concernée, les éventuels encadrements que j'ai mené pour ces travaux de recherche, leurs financements, ainsi que la liste des publications associées, triée par catégories. Les publications listées sont toutes postérieures à 2004, date de mon admission en tant de chargé de recherche permanent dans le laboratoire.

En vous souhaitant une bonne lecture !

Images et tenseurs

Sommaire

1.1 Définition d'une image	1
1.2 Dérivées premières et secondes directionnelles	2
1.3 Les tenseurs du second ordre	3



Ce court chapitre introduit les notations mathématiques couramment utilisées tout au long de ce manuscrit pour décrire les images et les champs tensoriels, ainsi que les calculs les plus usuels réalisés sur ces objets.

1.1 Définition d'une image

Avant même de parler d'algorithmes de traitement d'images, il est important de définir précisément ce que l'on entend par *image*. Dans un ordinateur, les images numériques sont stockées sous forme de données discrétisées, aussi bien spatialement (les pixels sont échantillonnés généralement sur une grille euclidienne régulière) que dans l'espace des valeurs des pixels (quantification). Cette discrétisation est souvent suffisamment fine pour permettre la considération de ces signaux discrets comme des fonctions mathématiques continues (ou au moins continues par morceaux), ce qui permet d'appliquer des outils mathématiques d'analyse fonctionnelle pour leur étude (aussi bien pour l'analyse d'images que pour la synthèse).

Les résultats probants obtenus par les nombreuses méthodes de la littérature, modélisant des images comme des fonctions continues, confirment depuis plusieurs années la vraisemblance de cette hypothèse, et c'est celle-ci que nous utiliserons par la suite.

On considère donc un domaine spatial $\Omega \subset \mathbb{R}^p$ dans un espace Euclidien de dimension p . Dans ce manuscrit, nous rencontrerons le cas $p = 2$ (images 2D usuelles), et $p = 3$ (images volumiques, ou séquences d'images).

Nous définissons alors une image \mathbf{I} comme une fonction (supposée continue) :

$$\mathbf{I} : \begin{cases} \Omega \rightarrow \mathbb{R}^n \\ \mathbf{X} \rightarrow \mathbf{I}(\mathbf{X}) \end{cases}$$

où $\mathbf{X} = x$ pour $p = 1$, $\mathbf{X} = (x, y)$ pour $p = 2$ et $\mathbf{X} = (x, y, z)$ pour $p = 3$.

$n \in \mathbb{N}^*$ représente le nombre de canaux de l'image \mathbf{I} . L'image sera dite *scalair*e lorsque $n = 1$,

et *multi-valuée* sinon ($n \geq 2$). Même si les valeurs de pixels des images numériques sont en pratique discrètes et délimitées, nous les modélisons de manière plus génériques comme prenant leurs valeurs dans l'espace continu \mathbb{R} .

Dans le cas des images multi-valuées, chaque point de l'image est un vecteur (ou une matrice) de dimension n . Typiquement, les images couleurs correspondent à $n = 3$ avec des valeurs vectorielles (R, G, B) (ou représentées dans un autre espace couleur, voir [POYNT095]). Nous notons dans ce cas par $I_i : \Omega \rightarrow \mathbb{R}$, la $i^{\text{ème}}$ composante vectorielle de \mathbf{I} , aussi appelée *canal de l'image* (avec $1 \leq i \leq n$). I_i est évidemment elle-même une image scalaire, et on a :

$$\forall \mathbf{X} \in \Omega, \quad \mathbf{I}_{(\mathbf{X})} = \left(I_{1(\mathbf{X})}, I_{2(\mathbf{X})}, \dots, I_{n(\mathbf{X})} \right)^T$$

En général, nous utiliserons des lettres **en gras** pour représenter des variables/images *multi-valuées*. Pour des raisons de concision, nous omettrons volontairement la coordonnées spatiale \mathbf{X} dans les expressions (\mathbf{I} au lieu de $\mathbf{I}_{(\mathbf{X})}$) quand cela sera possible sans ambiguïté.

1.2 Dérivées premières et secondes directionnelles

La dérivée d'une image scalaire I par rapport à la variable a sera notée

$$I_a = \frac{\partial I}{\partial a}$$

Les dérivations de I par rapport à toutes ses coordonnées spatiales \mathbf{X} donnent *le vecteur gradient*, symbolisé par ∇I :

$$\nabla I = (I_x, I_y)^T \quad \text{pour } p = 2, \quad \text{et} \quad \nabla I = (I_x, I_y, I_z)^T \quad \text{pour } p = 3.$$

L'ensemble des gradients de I forme un champ de vecteurs $\nabla I : \Omega \rightarrow \mathbb{R}^p$ significatif des orientations et des amplitudes des variations locales maximales des intensités lumineuses de I , en chaque point de Ω . L'image scalaire des *normes de gradient* $\|\nabla I\|$ est très fréquemment utilisée en analyse d'images, car elle donne une mesure du contraste local des structures présentes dans l'image, et donc de la présence ou non de contours.

$$\|\nabla I\| = \sqrt{I_x^2 + I_y^2 + I_z^2}$$

Plus généralement, la *dérivée directionnelle* d'une image scalaire I dans une direction \mathbf{u} normée de \mathbb{R}^p sera notée $I_{\mathbf{u}}$ et calculée de la façon suivante :

$$I_{\mathbf{u}} = \frac{\partial I}{\partial \mathbf{u}} = \nabla I \cdot \mathbf{u}$$

De manière similaire, on notera la dérivée seconde d'une image scalaire I par rapport à des variables a puis b :

$$I_{ab} = \frac{\partial^2 I}{\partial a \partial b}$$

et nous pouvons alors calculer le *hessien* de I comme la matrice \mathbf{H} des dérivées secondes par rapport aux coordonnées spatiales x et y (et z pour des images volumiques) :

$$\mathbf{H} = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{pmatrix} \quad \text{pour } p = 2, \quad \text{et} \quad \mathbf{H} = \begin{pmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{pmatrix} \quad \text{pour } p = 3.$$

La matrice \mathbf{H} est largement utilisée tout au long de ce manuscrit, notamment pour définir les processus de lissage/diffusion d'images. Et puisque nous considérons que nos images sont suffisamment régulières, $I_{xy} = I_{yx}$, et \mathbf{H} est donc une *matrice symétrique*. Nous utiliserons également fréquemment l'*opérateur Laplacien* Δ , défini comme

$$\Delta I = \text{trace}(\mathbf{H}) = I_{xx} + I_{yy} [+I_{zz}]$$

Il est également possible de calculer une *dérivée seconde directionnelle* de I dans une direction $\mathbf{u} \in \mathbb{R}^p$:

$$I_{\mathbf{uu}} = \frac{\partial^2 I}{\partial \mathbf{u}^2} = \nabla(\nabla I \cdot \mathbf{u}) \cdot \mathbf{u} = \mathbf{u}^T \mathbf{H} \mathbf{u} = \text{trace}(\mathbf{H} \mathbf{u} \mathbf{u}^T)$$

En pratique, cela revient à calculer, pour des images 2D et des images volumiques 3D :

$$\begin{cases} \text{pour } p = 2, & \mathbf{u} = (u, v), & I_{\mathbf{uu}} = u^2 I_{xx} + 2uv I_{xy} + v^2 I_{yy} \\ \text{pour } p = 3, & \mathbf{u} = (u, v, w), & I_{\mathbf{uu}} = u^2 I_{xx} + 2uv I_{xy} + 2uw I_{xz} + v^2 I_{yy} + 2vw I_{yz} + w^2 I_{zz} \end{cases}$$

Notons que pour les images multi-valuées \mathbf{I} , nous avons $\mathbf{I}_a(\mathbf{x}) \in \mathbb{R}^n$ et

$$\mathbf{I}_a = \left(\frac{\partial I_1}{\partial a}, \frac{\partial I_2}{\partial a}, \dots, \frac{\partial I_n}{\partial a} \right)^T \quad (1.1)$$

Les notions de gradient et de hessien ne se définissent pas aisément de manière ‘‘globale’’ pour une image multi-valorée \mathbf{I} , puisque chaque canal I_i possède son propre gradient et hessien (notés respectivement ∇I_i et \mathbf{H}_i). Il est par contre usuel d'étendre le concept de *norme de gradient* pour une image multi-valorée, en définissant la *norme de gradient multi-valorée* $\|\nabla \mathbf{I}\|$ suivante :

$$\|\nabla \mathbf{I}\| = \sqrt{\sum_{i=1}^n \|\nabla I_i\|^2} = \sqrt{\mathbf{I}_x \cdot \mathbf{I}_x + \mathbf{I}_y \cdot \mathbf{I}_y + [\mathbf{I}_z \cdot \mathbf{I}_z]} \quad (1.2)$$

où \mathbf{I}_x , \mathbf{I}_y et \mathbf{I}_z sont définis comme dans l'Eq.(1.1).

Comme dans le cas scalaire, l'image des $\|\nabla \mathbf{I}\|$ fournit en chaque pixel des mesures locales (scalaires) très intéressantes pour estimer en chaque point de \mathbf{I} les amplitudes de contraste local des vecteurs/couleurs (et donc la présence ou non de contours).

1.3 Les tenseurs du second ordre

Dans ce document, le terme *tenseur* est utilisé uniquement pour désigner une *matrice symétrique et semi définie-positve*.

Généralement, un *tenseur* désigne un objet mathématique plus complexe qu'une simple matrice (voir [ABRAHA12, SOKOLN51] pour des détails sur le calcul tensoriel généralisé), mais il est usuel en traitement d'images d'associer le mot *tenseur* au concept particulier de *tenseurs de diffusion* : Dans ce contexte, on se limite d'une part aux tenseurs d'ordre 2 (donc aux matrices) $n \times n$, et d'autre part aux matrices $\mathbf{T} \in \mathcal{P}(n)$ symétriques et semi définie-positives qui sont très bien adaptées à la représentation de *quantité de diffusion* suivant des *orientations spatiales* privilégiées.

Ce terme générique de *tenseur de diffusion* recouvre par exemple des modèles pour :

- La dispersion de l'erreur en calcul statistique (dans ce cas, les tenseurs sont aussi appelés *matrices de covariance*), voir par exemple [KANATA93, PENNEC06, LEEDAN00].
- Le comportement de lissage local de processus de régularisation. Ce point sera détaillé dans les chapitres 2 et 3.
- Ou encore, des lois de probabilité spatiale gaussienne, par exemple pour modéliser le mouvement des molécules d'eau dans les tissus biologiques. Ce type de tenseurs est par exemple fréquemment rencontré en imagerie *DT-MRI* (mentionné au chapitre 7).

Plus généralement, ce type particulier de tenseurs est apte à modéliser des *comportements diffusifs* quelconques, et on retrouve ces modèles dans de nombreuses applications autres qu'en traitement d'images. Cette habilitation cherche en partie à prouver la pertinence de ces modèles tensoriels pour diverses applications de traitement d'images.

Nous rappelons ici quelques propriétés mathématiques usuelles de ces tenseurs, à garder à l'esprit. Soit $\mathbf{T} = (t_{ij}) \in \mathbb{P}(n)$ un tenseur d'ordre 2 de taille $n \times n$. Alors,

$$\mathbf{T} \text{ est symétrique} \iff \forall i, j \in [1, n], \quad t_{ij} = t_{ji}$$

$$\mathbf{T} \text{ est semi définie-positif} \iff \forall \mathbf{X} \in \mathbb{R}^n, \quad \mathbf{X}^T \mathbf{T} \mathbf{X} \geq 0$$

Il faut se rappeler que les informations les plus significatives contenues dans un tenseur \mathbf{T} peuvent être récupérées à partir d'une *décomposition spectrale* qui donne ses vecteurs propres \mathbf{u}_k et ses valeurs propres λ_k (supposés ordonnés de manière décroissante, *c-à-d.* $k < l \Rightarrow \lambda_k \geq \lambda_l$).

Les éléments propres de \mathbf{T} vérifient les propriétés suivantes :

$$\mathbf{T} \text{ est semi définie-positif} \iff \forall k \in [1, n], \quad \lambda_k \geq 0$$

$$\mathbf{T} \text{ est à coefficients réels et symétrique} \iff \forall k, l \in [1, n], \quad \mathbf{u}_k \cdot \mathbf{u}_l = \delta_{kl} = \begin{cases} 1 & (\text{si } k = l) \\ 0 & (\text{si } k \neq l) \end{cases}$$

Les vecteurs propres \mathbf{u}_k forment ainsi une *base vectorielle orthonormée* de \mathbb{R}^n , et \mathbf{T} peut donc s'écrire sous la forme :

$$\mathbf{T} = \mathbf{R} \mathbf{\Gamma} \mathbf{R}^T \tag{1.3}$$

où $\mathbf{\Gamma} \in \mathbb{R}^{n \times n}$ est la matrice *diagonale* contenant les valeurs propres λ_k

$$\mathbf{\Gamma} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n) = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{pmatrix}$$

et \mathbf{R} est une *matrice de rotation* dont les colonnes $\tilde{\mathbf{u}}_k$ sont les vecteurs propres \mathbf{u}_k de \mathbf{T} , telles que le déterminant de \mathbf{R} soit $\det(\mathbf{R}) = +1$.

$$\mathbf{R} = (\tilde{\mathbf{u}}_1 \mid \tilde{\mathbf{u}}_2 \mid \dots \mid \tilde{\mathbf{u}}_n) \quad \text{où} \quad \forall k = 1..n, \quad \tilde{\mathbf{u}}_k = \pm \mathbf{u}_k$$

La formulation (1.3) sépare clairement les *orientations* \mathbf{R} des *diffusivités* $\mathbf{\Gamma}$ d'un tenseur \mathbf{T} . Une représentation graphique naturelle d'un tenseur \mathbf{T} est donc un ellipsoïde n -dimensionnel dont les axes et les rayons sont respectivement les vecteurs propres \mathbf{u}_k et les valeurs propres λ_k (positives) (Fig.1.1).

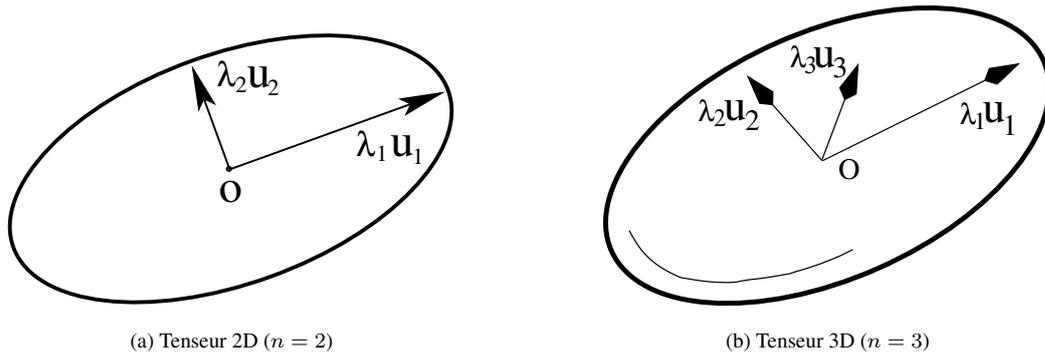


FIG. 1.1: Représentation d'un tenseur d'ordre 2 avec des ellipsoïdes.

Notons également que l'expression (1.3) peut se développer de la façon suivante :

$$\mathbf{T} = \sum_{k=1}^n \lambda_k \mathbf{u}_k \mathbf{u}_k^T$$

Cette seconde formulation met l'accent sur le fait qu'un tenseur \mathbf{T} peut être vu comme la somme pondérée de *tenseurs "élémentaires" orthogonaux* $\mathbf{u}_k \mathbf{u}_k^T$, orientés suivant une direction unique \mathbf{u}_k . Les n valeurs propres de ces tenseurs élémentaires $\mathbf{u}_k \mathbf{u}_k^T$ sont respectivement :

- 0, associé à $n - 1$ vecteurs propres \mathbf{v}_l formant une base orthonormée de l'hyperplan orthogonal à \mathbf{v}_k (donc $\forall l, \mathbf{v}_l \perp \mathbf{v}_k$).
- 1, associé au vecteur propre \mathbf{u}_k .

Ces tenseurs élémentaires $\mathbf{u}_k \mathbf{u}_k^T$ peuvent être imaginés comme des ellipsoïdes *minces* ayant un axe de rayon 1, orienté suivant \mathbf{u}_k et tous les autres de rayon 0. Ils sont particulièrement utiles pour représenter uniquement l'information d'*orientation* des vecteurs \mathbf{u}_k , sans notion de direction. Un tenseur générique \mathbf{T} est simplement une combinaison pondérée d'orientations orthogonales.

Quand les n valeurs propres λ_k d'un tenseur \mathbf{T} sont toutes égales à λ , nous sommes dans le cas où le tenseur ne possède pas d'*orientation privilégiée*. Le tenseur est alors dit *isotrope*, et correspond à une version pondérée de la matrice identité :

$$\mathbf{T} = \sum_{k=1}^n \lambda \mathbf{u}_k \mathbf{u}_k^T = \lambda \mathbf{R} \mathbf{R}^T = \lambda \mathbb{I}_d$$

La représentation graphique d'un tenseur *isotrope* est donc un *disque* (ou une *sphere*) de rayon λ , et un processus de diffusion correspondant serait réalisé avec la même amplitude dans toutes les directions de l'espace. Dans le cas contraire (au moins une orientation du tenseur est dominante), le tenseur est dit *anisotrope*. Plus l'écart entre la valeur propre principale λ_1 (associé au vecteur propre principal \mathbf{u}_1) et les autres valeurs propres $\lambda_{k>1}$ est grand, plus le tenseur est *anisotrope* (ellipsoïde de forme allongée, comme celui représenté en Fig.1.1).

Une autre propriété intéressante du tenseur d'ordre 2 devient visible lorsque l'on considère celui-ci comme une application linéaire agissant sur un vecteur $\mathbf{X} \in \mathbb{R}^n$ (*c-à-d.* en calculant $\mathbf{X}' = \mathbf{T}\mathbf{X}$). Un tenseur \mathbf{T} peut en effet être vu comme un opérateur de *déformation*, qui va modifier l'orientation et l'amplitude d'un vecteur \mathbf{X} du plan de façon à le ramener plus ou moins vers

une orientation unique, selon le degré d'anisotropie du tenseur considéré (calculé par exemple pour des tenseurs 2×2 par $Anisotropie = (\lambda_1 - \lambda_2)/(\lambda_1 + \lambda_2)$).

C'est ce qu'illustre la Fig.1.2 ci dessous. En haut à gauche ($Anisotropie=0$), on affiche les vecteurs $\mathbf{X} \in \mathbb{R}^2$ correspondant à différentes orientations du plan, avec une teinte dépendante de l'orientation de chaque vecteur. Lorsque l'anisotropie du tenseur \mathbf{T} est nulle, \mathbf{T} est équivalent à la matrice identité (à un facteur près), et les orientations des vecteurs $\mathbf{X}' = \lambda\mathbf{X}$ ne sont donc pas modifiées. Au fur et à mesure que l'anisotropie du tenseur \mathbf{T} augmente, les vecteurs $\mathbf{X}' = \mathbf{T}\mathbf{X}$ deviennent de plus en plus attirés vers le vecteur propre principal \mathbf{u}_1 de \mathbf{T} , et leur norme $\|\mathbf{X}'\|$ diminue d'autant que \mathbf{X} et \mathbf{u}_1 n'ont pas la même orientation.

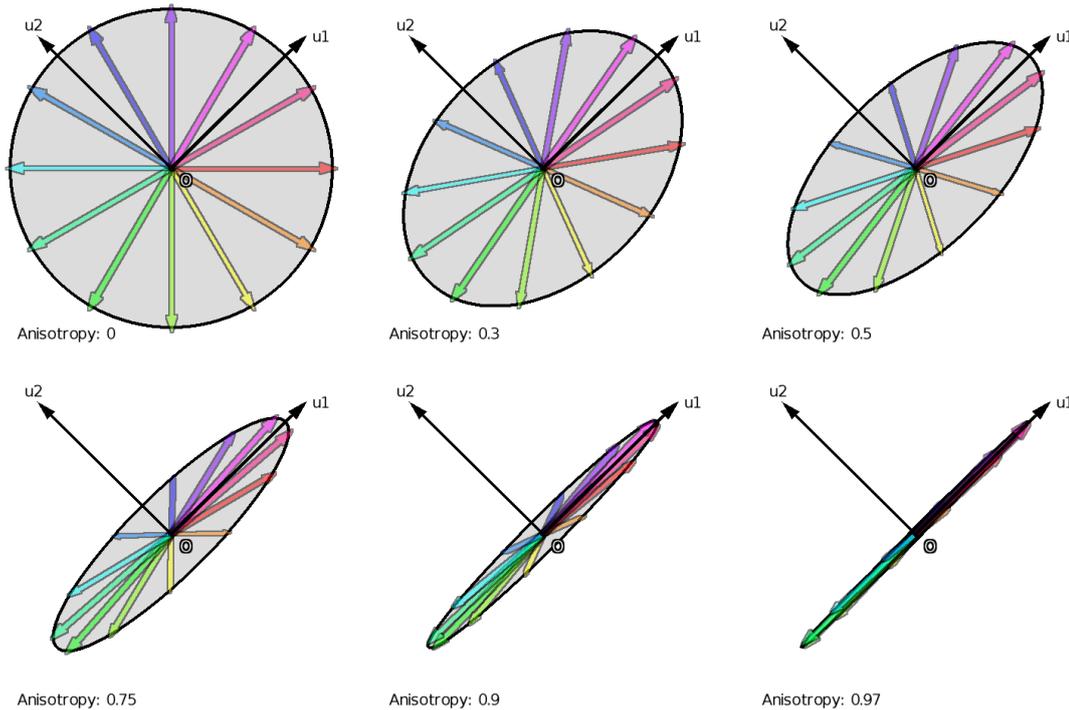


FIG. 1.2: Propriété de déformation de vecteurs par un tenseur

Dans le cas extrême où l'anisotropie du tenseur \mathbf{T} vaut 1, *c-à-d.* que $\mathbf{T} = \lambda_1 \mathbf{u}_1 \mathbf{u}_1^T$, chaque orientation originale \mathbf{X} est transformée en $\mathbf{X}' = \lambda_1 (\mathbf{u}_1 \cdot \mathbf{X}) \mathbf{u}_1$, qui est donc tout le temps orienté suivant \mathbf{u}_1 (sauf dans le cas $\mathbf{u}_1 \perp \mathbf{X}$ où \mathbf{X}' devient nul).

Dans les chapitres suivants, nous utiliserons cet ensemble non-exhaustif de propriétés mathématiques pour l'analyse et la résolution des problèmes traités. Davantage de détails sur la théorie des tenseurs et certaines de ses applications pour l'image peuvent être trouvés dans les excellents ouvrages [GRANLU95, MEDION00].

Lissage anisotrope d'images guidé par champ de tenseurs

Sommaire

2.1 Introduction au lissage d'images	7
2.2 Géométrie locale des images multi-canaux	10
2.3 Lissage d'images multi-canaux par EDP	15
2.4 Conclusion du chapitre	24



Nous dressons ici un état de l'art des méthodes de la littérature pour le lissage et la régularisation d'images par équations aux dérivées partielles (EDP) de diffusion anisotrope. Les techniques de lissage d'images par EDP ont en effet connu un regain d'intérêt incontestable à partir du moment où des modèles tensoriels de géométrie des images ont été introduits et pris en compte dans les équations de diffusion. C'était l'un des points clé étudié pendant ma thèse de doctorat (1999–2002). Beaucoup d'idées et de concepts présentés ici ont naturellement inspiré les méthodes et les algorithmes élaborés par la suite, et qui sont décrits dans les chapitres suivants.

Période : 2000–2007.

Publications associées :

– Chapitres de livre : [BC6], [BC5], [BC3], [BC2].

2.1 Introduction au lissage d'images

Dans le domaine du traitement d'images, l'utilisation de champs tensoriels a été largement popularisé par les nombreuses méthodes basées EDP, proposées dans la littérature pour tenter de résoudre le problème de la régularisation de données image. Cette problématique de régularisation a toujours été centrale en traitement d'images : d'une part, la régularisation permet d'*atténuer le bruit* présent dans les données et constitue donc une étape quelquefois incontournable de pré-traitement, aboutissant à une meilleure performance des algorithmes d'analyse de plus haut niveau, par exemple pour la détection de caractéristiques spécifiques dans les images (contours, coins, objets, mouvement, etc.). D'autre part, la régularisation permet de créer des *versions simplifiées* des images, et joue donc un rôle important dans le cadre de l'analyse multi-échelle des objets contenus dans celle-ci. D'une manière plus générale, la régularisation joue un

rôle fondamental pour trouver des solutions cohérentes à des problèmes de traitement d'images initialement *mal posés* [HANDAM23], comme les problèmes de restauration, de segmentation, de recalage, de reconstruction de surfaces, pour n'en citer que quelques uns. Cela explique le fort intérêt que la communauté du traitement d'images a toujours eu vis à vis de ce sujet, et pourquoi tant de formalismes de régularisation différents ont pu voir le jour dans la littérature.

Comment les champs tensoriels sont-ils apparus dans les méthodes de régularisation d'images ? Nous proposons ici un petit historique retraçant l'arrivée progressive des tenseurs de diffusion dans les algorithmes classiques de lissage anisotropes d'images par EDP, algorithmes efficaces qui sont capables de lisser les images tout en préservant leurs discontinuités locales (contours et textures).

2.1.1 Historique des EDP pour le lissage d'images

Perona & Malik, dans leurs travaux pionniers du début des années 90 [PERONA90], ont été les premiers à imaginer la régularisation d'images en terme d'évolution d'EDP de diffusion anisotrope. Leur méthode a initié l'intérêt particulier apporté aux formulations EDP, car elle était capable de lisser les données d'une manière totalement *non linéaire*, d'une part, en supprimant le bruit de manière satisfaisante, mais surtout, en autorisant la préservation des discontinuités significatives dans les images, comme les contours et les coins (c'était d'autant plus remarquable qu'il a été démontré plus tard que la formulation mathématique proposée était intrinsèquement instable [KICHEN97, WEICKE97B]). Les EDP de diffusion, apparues initialement pour modéliser l'évolution locale de systèmes physiques complexes (liquide, gaz), sont des objets mathématiques très étudiés, et de nombreux résultats théoriques les concernant ont eu des répercussions pour la régularisation de données. Dans le domaine du traitement d'images, une EDP est un modèle idéal pour décrire des *interactions locales* entre pixels et peut donc facilement s'adapter pour traiter le cas d'images corrompues par des sources de dégradation qui sont elles-mêmes locales ou semi-locales, ce qui n'est en rien restrictif : le bruit, les rayures fines, les artefacts de compression, sont par exemple des dégradations locales très souvent rencontrées en imagerie numérique. En suivant la voie ouverte par Perona & Malik, de nombreux auteurs ont élaboré, par la suite, des variantes d'EDP de diffusion pour la régularisation d'images, notamment pour les images à valeurs scalaires. Les contributions théoriques importantes dans ce domaine ont porté sur la façon dont l'équation de diffusion isotrope classique (la plus simple, c-à-d l'équation de la chaleur) peut être étendue pour se transformer en lissage anisotrope [PERONA90, SAPIRO01, WEICKE98], sur la façon dont les EDP de diffusion peuvent être vues comme des descentes de gradient de fonctionnelles variées [AUBERT02, CHAMBO97, BLANC-95, KIMMEL00, RUDIN92], et le lien entre les EDP de régularisation et le concept d'espace d'échelle non linéaire [ALVARE93, LINDEB13, NIELSE97]. Les extensions de ces techniques pour traiter le cas des images couleur, et plus généralement des données multi-dimensionnelles sont apparues ensuite [BLOMGR98, KIMMEL00, SAPIRO01, SAPIRO96B, TSCHUM05, WEICKE99B], amenant à des expressions faisant intervenir un ou plusieurs termes de couplage entre les différents canaux des images considérées. Des équations de diffusion traitant de cas multi-valués *constraints* ont également été proposées, permettant par exemple la régularisation de champ de vecteurs unitaires [KIMMEL02, PERONA98, TANG98], de matrices orthonormées [CHEFD'04], de matrices définie-positives [CHEFD'04, TSCHUM01], ou encore de données image définies sur des surfaces implicites [BERTAL01B, SHEN01, TANG00]. En pratique, ces EDP de diffu-

sion spécifiques possèdent un terme additionnel de préservation de la contrainte, qui s'ajoute au terme de régularisation.

2.1.2 Principe de fonctionnement

Malgré le nombre important de formalismes EDP existants pour la régularisation d'images scalaires ou multi-canaux, on peut remarquer qu'ils utilisent tous un même principe commun : une EDP de régularisation non-linéaire $\frac{\partial I}{\partial t} = \mathcal{R}$ a une action de *lissage local* de l'image I , le long d'une ou plusieurs orientations du plan, choisies différemment en chaque point de l'image, et dépendantes idéalement de la configuration locale des pixels avoisinants. Typiquement, la direction de lissage privilégiée est choisie pour être parallèle aux contours des structures présentes dans les images, résultant en une régularisation *anisotrope* qui ne détruit pas ces contours. Cela a une interprétation intéressante en terme d'*espace d'échelle* : tandis que les données image sont régularisées petit à petit (suivant la variable de temps t d'évolution de l'EDP), une séquence d'images $I^{[t]}$ de plus en plus lisses est générée. De ce point de vue, il semble donc naturel que les algorithmes de régularisation fassent disparaître d'abord les caractéristiques les moins significatives des images (dans l'idéal : le bruit, en pratique : les données les moins contrastées), pendant que les détails intéressants de l'image (les contours) restent présents jusqu'à ce qu'ils deviennent eux-même négligeables par rapport aux autres discontinuités restantes (c-à-d des contours encore plus contrastés) [ALVARE93, LINDEB13, NIELSE97, PERONA90, WITKIN83]. D'un point de vue géométrique, les EDP de régularisation peuvent donc être vues comme des filtres itératifs et localement adaptatifs, qui simplifient l'image petit à petit en minimisant les variations de niveaux dans les voisinages des pixels (Fig.2.1). Il en résulte que la plupart des équations

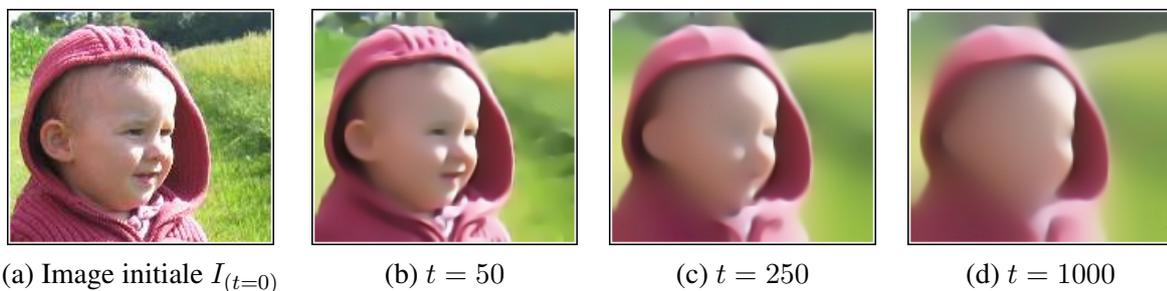


FIG. 2.1: EDP de régularisation non-linéaire et notion d'espace d'échelle anisotrope.

de lissage convergent vers une solution constante (à $t \rightarrow \infty$), qui n'est pas très intéressante, mais qui minimise effectivement les variations de niveaux dans l'image résultante. Pour éviter cette sur-simplification des image régularisées, on considère quelquefois une vitesse d'évolution d'EDP de type $\mathcal{R}' = \mathcal{R} + \alpha (I_{\text{noisy}} - I)$ contenant un terme additionnel d'*attache aux données* (pondéré par un paramètre $\alpha \in \mathbb{R}^+$ défini par l'utilisateur), permettant à l'image régularisée de ne pas être trop éloignée de l'image originale bruitée (donc non constante). Une autre technique possible est de stopper le flot de régularisation pur $\frac{\partial I}{\partial t} = \mathcal{R}$ après un nombre fini d'itérations (qui devient donc un paramètre de la méthode). Dans ce chapitre, nous nous focaliserons sur l'étude des termes de régularisation \mathcal{R} seuls. Pour une étude avancée sur les termes d'attache aux données linéaires et/ou non-linéaires, on se reportera aux travaux de [NIKOLO01].

Orienter le lissage local est donc l'une des idées clés sur laquelle se basent la plupart des méthodes de régularisation par EDP. Le problème se pose alors de la définition d'une *géométrie*

locale cohérente des images scalaires ou multi-canaux. Ce doit être l'une des premières préoccupations d'un bon algorithme de régularisation. C'est en suivant ce principe que les auteurs de [TSCHUM05, WEICKE98] ont proposé deux formalismes différents et génériques, capables d'élaborer des processus de régularisation dont la géométrie de lissage peut être finement choisie suivant l'application considérée. Ces méthodes ont deux intérêts majeurs : d'une part, elles unifient un grand nombre d'équations de diffusion proposées précédemment dans la littérature en donnant une interprétation géométrique locale des processus de régularisation correspondants. D'autre part, elles séparent clairement *la définition de la géométrie de lissage* du processus de lissage lui-même : dans un premier temps, on analyse la géométrie des structures locales des images. Puis, on définit la géométrie locale du lissage souhaité au moyen d'un champ de *tenseurs de diffusion* qui est déduit de l'analyse précédente. Pour finir, une étape du processus de lissage proprement dit est effectuée par le biais d'une ou plusieurs itérations d'une EDP de diffusion spécifique. Cette procédure est répétée jusqu'à l'obtention d'une image suffisamment régularisée.

Dans ce chapitre, nous résumerons d'abord les choix possibles pour définir des géométries locales cohérentes pour les images multi-canaux, en proposant un état de l'art et en comparant les diverses solutions de la littérature [BLOMGR98, DI ZEN86, SAPIRO96A, WEICKE98] (section 2.2). Cela nous amènera naturellement à l'utilisation des tenseurs d'ordre 2 pour une modélisation optimale de la géométrie locale des images.

2.2 Géométrie locale des images multi-canaux

2.2.1 Quelles caractéristiques géométriques ?

Un processus de régularisation d'images peut être vu principalement comme l'application d'un filtre de lissage qui réduit localement les variations des niveaux des pixels. Plus précisément, nous souhaitons lisser une image multi-canaux $\mathbf{I} : \Omega \rightarrow \mathbb{R}^n$ tout en préservant ses contours (les discontinuités du signal), c-à-d effectuer un lissage localement adaptatif principalement le long de la direction des contours et éviter au maximum un lissage orthogonal à ceux-ci. Dans un premier temps, une idée naïve serait d'appliquer un filtre de régularisation à valeur scalaire sur chaque composante I_i d'une image multi-canaux \mathbf{I} , en le faisant indépendamment pour chaque canal $i = 1 \dots n$. Dans ce cas, la corrélation possible qui existe entre les différentes composantes des images serait ignorée, avec pour conséquence, des disparités importantes dans les comportements locaux de lissage entre chaque canal : les directions et les amplitudes de lissage pourraient être très différentes d'une composante à l'autre. De telles stratégies de régularisation amènent classiquement des effets de sur-lissage indésirables qui peuvent détruire les structures des contours de manière rapide.

Une méthode de régularisation d'images multi-canaux devrait plutôt se baser sur une stratégie de lissage inter-composante cohérente, de telle manière que les directions et les amplitudes de lissage soient localement les mêmes pour tous les canaux I_i . Cela signifie qu'il faut tout d'abord obtenir une mesure robuste de la *géométrie locale multi-valorée* d'une image \mathbf{I} . Pratiquement, une telle géométrie peut se définir par les caractéristiques suivantes, et ce, en tout point $\mathbf{X} = (x, y)$ de \mathbf{I} :

1. Deux directions orthogonales $\theta_{+(\mathbf{X})}$, $\theta_{-(\mathbf{X})} \in S^1$, dirigées respectivement orthogonalement et parallèlement aux contours. Elles sont choisies généralement comme étant les

directions de variations maximales et minimales des niveaux d'intensités de l'image en \mathbf{X} . θ_- correspond alors à la *direction du contour*, quand il y en a un, tandis que θ_+ s'assimile naturellement à la notion de *direction de gradient*.

2. Une mesure de variation $\mathcal{N}_{(\mathbf{X})}$ qui mesure le *contraste local* d'un contour. Dans le cas des images multi-canaux, cette mesure étend le concept de *norme du gradient*, utilisé habituellement dans le cas des images scalaires.

Pour définir ces caractéristiques, différentes approches ont été considérées dans la littérature et sont présentées ci-dessous. Pour des raisons de lisibilité, nous omettrons par la suite d'indiquer le suffixe (\mathbf{x}) dans les expressions des fonctions dont la valeur est manifestement considérée au point $\mathbf{X} = (x, y)$.

2.2.2 Géométrie estimée à partir d'une caractéristique scalaire

Une méthode très simple d'estimation géométrique consisterait à réduire dans un premier temps une image multi-canaux en une image scalaire $f(\mathbf{I})$, qui idéalement, contiendrait toute l'information nécessaire sur les contours. La fonction de conversion $f : \mathbb{R}^n \rightarrow \mathbb{R}$ pourrait s'inspirer des modèles mathématiques de la perception humaine des contours dans les images, ce qui est tout à fait concevable pour le cas particulier des images couleur : on peut choisir par exemple la fonction f comme étant la luminosité (réponse perceptuelle à la luminance) telle que définie dans l'espace de représentation couleur *CIE - Lab* [POYNTO95] :

$$f = L^* = 116 g(Y) - 16 \quad \text{avec} \quad Y = 0.2125R + 0.7154G + 0.0721B$$

où $g : \mathbb{R} \rightarrow \mathbb{R}$ est défini par

$$\begin{cases} g(s) = \sqrt[3]{s} & \text{si } s > 0.008856 \\ g(s) = 7.787s + \frac{16}{116} & \text{sinon} \end{cases} \quad (2.1)$$

La géométrie locale couleur $\{\mathcal{N}, \theta_+, \theta_-\}$ de \mathbf{I} peut alors se définir par :

$$\begin{cases} \theta_+ = \frac{\nabla f(\mathbf{I})}{\|\nabla f(\mathbf{I})\|} \\ \theta_- \perp \theta_+ \end{cases} \quad \text{et} \quad \mathcal{N} = \|\nabla f(\mathbf{I})\| \quad (2.2)$$

Cette technique a deux inconvénients majeurs. D'abord, il n'est pas toujours possible de définir une fonction f significative pour n'importe quel type d'images multi-canaux (notamment quand le nombre de canaux excède $n > 3$). Ensuite, parce qu'il n'existe mathématiquement pas de fonction scalaire $f(\mathbf{I})$ qui serait capable de détecter toutes les variations multi-valuées d'une image \mathbf{I} . Par exemple, la fonction luminosité définie précédemment sera inapte à détecter des contours d'*iso-luminosité* dans une image couleur, comme illustrée en Fig.2.2 : les contours à l'intérieur du symbole du Yin/Yang ne peuvent être détectés par $\mathcal{N} = \|\nabla f(\mathbf{I})\|$ puisque la luminosité $f(\mathbf{I})$ des deux couleurs a été choisie à l'identique. Un lissage se basant sur cette géométrie aura probablement un comportement local isotrope, ou orienté dans une direction erronée, amenant dans les deux cas à un sur-lissage (donc une destruction) de ces contours.

2.2.3 Géométrie multi-valuée de Di Zenzo

Pour aller au delà de cette limitation, une solution élégante a été proposée par Di Zenzo dans [DI ZEN86]. Il considère une image multi-canaux $\mathbf{I} : \Omega \rightarrow \mathbb{R}^n$ comme un champ de vecteurs,

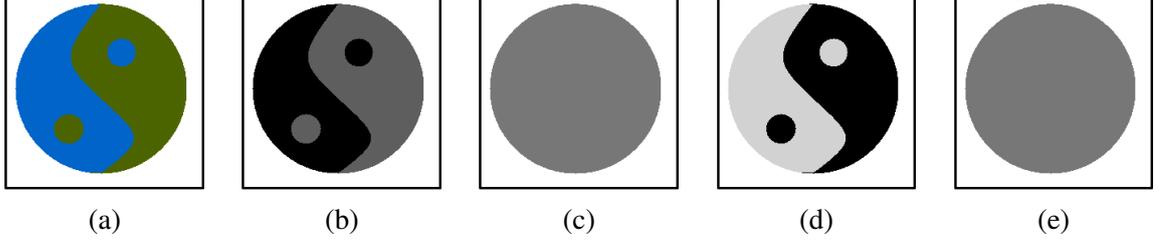


FIG. 2.2: L'utilisation d'une fonction scalaire pour définir une géométrie locale couleur peut échouer à détecter certains contours. (a) Image couleur, (b) Composante rouge R , (c) Composante verte G , (d) Composante bleue B , (e) Image scalaire de luminosité L^* .

et s'intéresse à la norme euclidienne $\|d\mathbf{I}\|^2$ de la variation infinitésimale vectorielle locale $d\mathbf{I}$ en tout point $\mathbf{X} \in \Omega$:

$$d\mathbf{I} = \mathbf{I}_x dx + \mathbf{I}_y dy \quad \text{où} \quad \mathbf{I}_x = \frac{\partial \mathbf{I}}{\partial x} \quad \text{et} \quad \mathbf{I}_y = \frac{\partial \mathbf{I}}{\partial y} \quad (\in \mathbb{R}^n)$$

et donc, $\|d\mathbf{I}\|^2 = d\mathbf{I}^T d\mathbf{I} = \|\mathbf{I}_x\|^2 dx^2 + 2 \mathbf{I}_x^T \mathbf{I}_y dx dy + \|\mathbf{I}_y\|^2 dy^2$, c-à-d.

$$\|d\mathbf{I}\|^2 = d\mathbf{X}^T \mathbf{G} d\mathbf{X} \quad \text{où} \quad \mathbf{G} = \sum_{i=1}^n \nabla I_i \nabla I_i^T \quad \text{et} \quad d\mathbf{X} = \begin{pmatrix} dx \\ dy \end{pmatrix}, \quad \nabla I_i = \begin{pmatrix} I_{ix} \\ I_{iy} \end{pmatrix}$$

La matrice \mathbf{G} est nommée *tenseur de structure*. Elle moyenne les contributions de variations de niveaux de chaque canal I_i de l'image \mathbf{I} . C'est une matrice 2×2 symétrique et semi définie-positive, dont les coefficients sont :

$$g_{11} = \sum_{i=1}^n I_{ix}^2, \quad g_{12} = g_{21} = \sum_{i=1}^n I_{ix} I_{iy}, \quad \text{et} \quad g_{22} = \sum_{i=1}^n I_{iy}^2$$

Di Zenzo préconise d'estimer les ∇I_i par un filtre de Deriche [ALVARE00]. Dans le cas usuel des images couleur $\mathbf{I} = (R, G, B)$, le tenseur de structure \mathbf{G} se résume donc en :

$$\mathbf{G} = \begin{pmatrix} R_x^2 + G_x^2 + B_x^2 & R_x R_y + G_x G_y + B_x B_y \\ R_x R_y + G_x G_y + B_x B_y & R_y^2 + G_y^2 + B_y^2 \end{pmatrix} \quad (2.3)$$

Il est intéressant de noter que les deux valeurs propres positives $\lambda_{+/-}$ de \mathbf{G} mesurent respectivement les valeurs maximales et minimales de la *norme de variation* vectorielle $\|d\mathbf{I}\|^2$, tandis que les vecteurs propres orthogonaux associés θ_+ et θ_- donnent les *orientations* correspondantes de ces extrema. Ils sont définis analytiquement par :

$$\lambda_{+/-} = \frac{g_{11} + g_{22} \pm \sqrt{\delta}}{2} \quad \text{et} \quad \theta_{+/-} \parallel \begin{pmatrix} 2 g_{12} \\ g_{22} - g_{11} \pm \sqrt{\delta} \end{pmatrix} \quad (2.4)$$

où $\delta = (g_{11} - g_{22})^2 + 4 g_{12}^2$. Par la suite, on considère que les vecteurs θ_{\pm} ont été normalisés en vecteurs unitaires.

Avec cette approche simple, Di Zenzo a ouvert une voie naturelle pour analyser la géométrie vectorielle locale des images multi-canaux, par l'utilisation de la base orthonormée d'orientation $\theta_+ \perp \theta_-$ et de ses mesures de contrastes associés λ_+, λ_- . Une légère variante de ce tenseur de

structure a été proposée par Weickert dans [WEICKE98]. Celui-ci propose d'étudier plutôt les éléments spectraux de $\mathbf{G}_{\alpha,\sigma}$, version doublement lissée du tenseur de structure \mathbf{G} :

$$\mathbf{G}_{\alpha,\sigma} = \sum_{i=1}^n [(\nabla I_{i_\alpha} \nabla I_{i_\alpha}^T) * G_\sigma] \quad \text{où} \quad I_{i_\alpha} = I_i * G_\alpha \quad (2.5)$$

où G_α et G_σ sont des noyaux gaussiens 2D d'écart types respectifs α et $\sigma \in \mathbb{R}^+$. Ici, on applique les convolutions séparément sur chaque coefficient du tenseur brut \mathbf{G} . Les deux paramètres α et σ permettent d'introduire une contrainte de régularité sur le champ de tenseurs de structure calculé, et par extension, de contraindre la régularité de la géométrie locale multi-canaux de l'image considérée.

L'analyse des valeurs propres λ_+ , λ_- de $\mathbf{G}_{\alpha,\sigma}$ permet de discriminer différentes configurations géométriques locales des images :

- Quand $\lambda_+ \approx \lambda_- \approx 0$, il y a très peu de variations vectorielles autour du point \mathbf{X} courant : le voisinage est *pratiquement constant* et ne contient probablement ni contours, ni coins (intérieur des bandes de la Fig.2.3a). Pour cette configuration, la norme de variation \mathcal{N} que nous devons définir devrait être faible, voire nulle.
- Quand $\lambda_+ \gg \lambda_-$, il y a des variations vectorielles significatives à proximité du point courant, qui est probablement situé sur un *contour*, ou un *coin* (contours des bandes de la Fig.2.3a). Pour cette configuration, la norme de variation \mathcal{N} à définir devrait être idéalement élevée.
- Quand $\lambda_+ \approx \lambda_- \gg 0$, le point courant est localisé sur un *point selle de la surface vectorielle*, ce qui représente une structure de coin particulière dans l'image (intersections des bandes de la Fig.2.3a). Ici, la valeur de \mathcal{N} devrait être idéalement encore plus élevée que pour les configurations précédentes, car cette forme complexe de type "jonctions" est habituellement difficile à lisser sans être dégradée. On souhaite donc atténuer d'autant plus le lissage ici, en favorisant une mesure de variation très forte (ce qui revient à dire qu'un coin est assimilable à un contour très contrasté).

En pratique, beaucoup d'algorithmes de régularisation d'images multi-canaux proposés dans la littérature se sont implicitement ou explicitement basés sur ces attributs géométriques de Di Zenzo. Plus particulièrement, trois choix distincts de normes de variations vectorielles \mathcal{N} ont été considérées :

1. $\mathcal{N} = \sqrt{\lambda_+}$, comme l'extension la plus naturelle de la norme de gradient des images scalaires, vue comme *la valeur de variation locale maximale* [BLOMGR98, SAPIRO96A, SAPIRO97] (Fig.2.3b). Cette norme ne donne cependant pas particulièrement d'importance aux configurations de type coins par rapport aux contours rectilignes.
2. $\mathcal{N}_- = \sqrt{\lambda_+ - \lambda_-}$, également dénommée *mesure de cohérence* [SAPIRO96B, WEICKE98]. Notons que cette mesure ne permet malheureusement pas de détecter correctement les discontinuités qui sont des points selles des images multi-canaux, vues comme des surfaces vectorielles "d'élévation". Ce fait est particulièrement visible aux intersections des bandes, et au centre et aux parties gauche et droite de l'œil de l'enfant, dans la Fig.2.3c. C'est une mesure à éviter dans le cadre d'un processus de régularisation, puisque les régions contenant des coins vont être traitées au même titre que les régions homogènes (\mathcal{N}_- sera faible dans les deux cas), alors que l'on souhaite évidemment lisser localement ces deux configurations géométriques de manière très différente.
3. $\mathcal{N}_+ = \sqrt{\lambda_+ + \lambda_-}$, parfois dénotée $\|\nabla \mathbf{I}\|$ est souvent choisie [BLOMGR98, TANG98, TSCHUM01], car elle permet de détecter à la fois les contours et les coins d'une façon

satisfaisante, tout en étant facilement calculable. Elle ne requiert en effet aucune décomposition spectrale de $\mathbf{G}_{\alpha,\sigma}$, puisque $\mathcal{N}_+ = \sqrt{\text{trace}(\mathbf{G})} = \sqrt{\sum_{i=1}^n \|\nabla I_i\|^2}$. Cette mesure donne plus de poids aux points selles de l'image multi-canaux (Fig.2.3d), ce qui va permettre d'atténuer avantageusement le lissage sur les structures de type coin, dans le cadre du débruitage d'images.

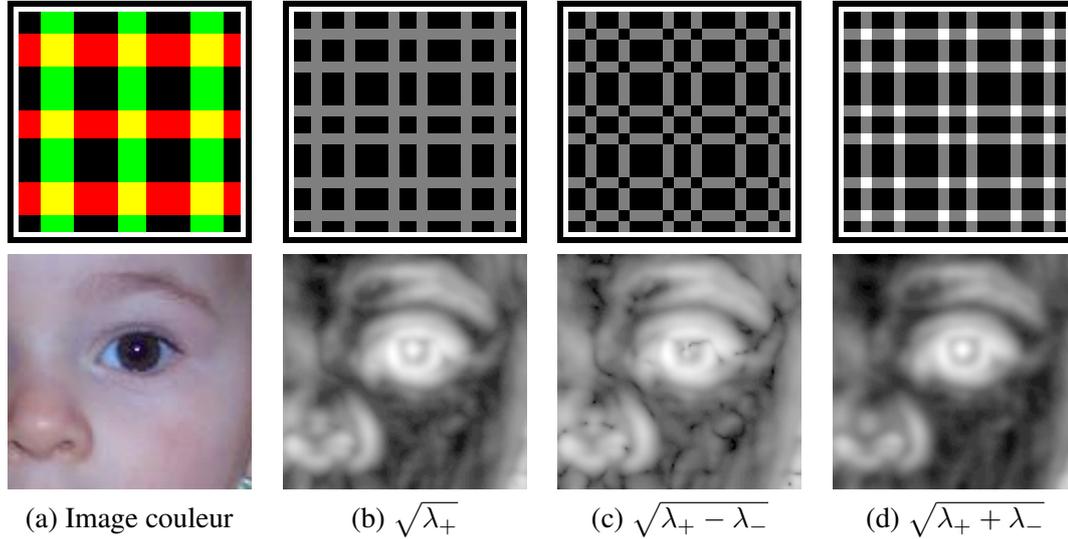


FIG. 2.3: Comparaison des mesures locales de variations \mathcal{N} , \mathcal{N}_- et \mathcal{N}_+ sur deux images couleur (les pixels clairs ont les valeurs les plus élevées).

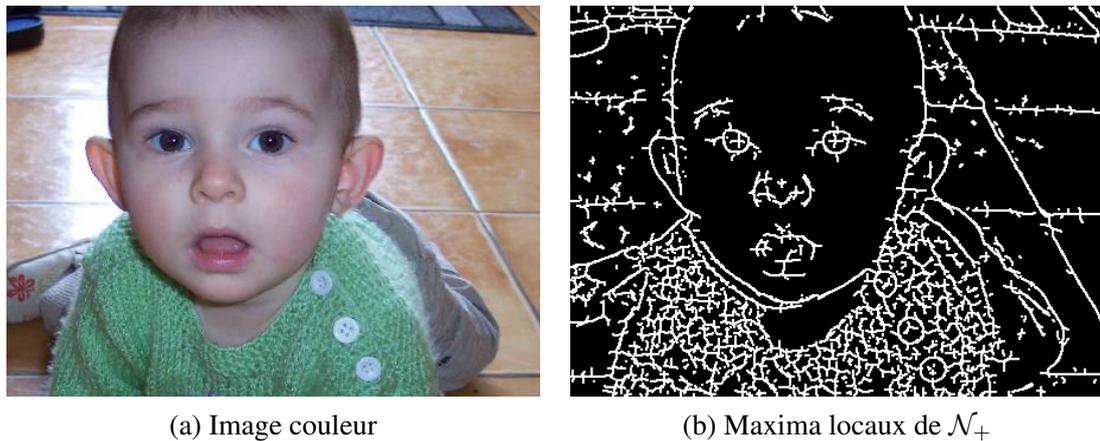


FIG. 2.4: Calcul des extrema de la mesure \mathcal{N}_+ pour la détection de contours couleurs.

Il est également intéressant de remarquer que dans le cas scalaire ($n = 1$), le tenseur de structure se réduit en :

$$n = 1 \implies \|dI\|^2 = d\mathbf{X} \mathbf{G}^1 d\mathbf{X} \quad \text{où} \quad \mathbf{G}^1 = \nabla I \nabla I^T = \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix}$$

Et dans ce cas, les vecteurs propres $\theta_{+/-}^1$ et les valeurs propres $\lambda_{+/-}^1$ de \mathbf{G}^1 sont :

$$\begin{cases} \theta_-^1 = \xi = \frac{\nabla I^\perp}{\|\nabla I\|} \\ \theta_+^1 = \eta = \frac{\nabla I}{\|\nabla I\|} \end{cases} \quad \text{associés à} \quad \begin{cases} \lambda_-^1 = 0 \\ \lambda_+^1 = \|\nabla I\|^2 \end{cases}$$

Dans le cas des images scalaires, les trois mesures de variations \mathcal{N}_+ , \mathcal{N}_- et \mathcal{N} se réduisent alors toutes en $\|\nabla I\|$, ce qui est une propriété naturelle et désirable.

Une fois que la géométrie locale vectorielle a été définie, nous pouvons l'utiliser pour mesurer toute configuration d'image locale en traitement d'images multi-canaux, et ce, pour tout type d'applications (non limitées à la régularisation). Par exemple, la détection de contours couleurs peut être réalisée en détectant les maxima locaux de la mesure de contraste local \mathcal{N}_+ (Fig.2.4 et [KOSCHA95, TSCHUM02]). De la même façon, cette analyse géométrique vectorielle locale de Di Zenzo a été intégrée pour la mesure des contours dans certaines méthodes de segmentation d'images multi-canaux par EDP [SAPIRO96A, SAPIRO97]. Dans le cadre de la régularisation d'images, la mesure $\mathcal{N}_+ = \sqrt{\lambda_+ + \lambda_-}$ est la mesure la plus adaptée pour détecter les variations locales de contraste dans des images multi-canaux.

2.3 Lissage d'images multi-canaux par EDP

Nous proposons une revue des méthodes classiques de lissage d'images multi-canaux basées sur les EDP de diffusion anisotropes, et une classification de ces méthodes suivant trois différents axes, reliés à des niveaux d'interprétation distincts du processus de régularisation, du plus global au plus local. Pour chaque section, nous décrirons d'abord l'idée initiale, élaborée pour le cas des images scalaires, puis son extension pour des données image multi-canaux.

2.3.1 Méthodes variationnelles

Au contraire de la formulation directe de Perona-Malik, de nombreuses méthodes de la littérature ont proposé une vision variationnelle de la régularisation, en la considérant comme une procédure de minimisation d'énergie globale à l'image. Les approches décrites notamment dans [AUBERT02, CHAMBO97, BLANC-95, SAPIRO01, WEICKE98] ont contribué à définir diverses fonctionnelles mesurant des variations globales d'images. Intuitivement, minimiser ce type de fonctionnelles va permettre de réduire les variations de niveaux (donc enlever graduellement le bruit), tout en prenant garde à préserver les variations importantes (pour éviter la destruction des contours). La formulation dite des ϕ -fonctionnelles regroupe la plupart de ces approches dans un formalisme commun élégant.

Une image *scalair*e bruitée I_{bruitee} peut être régularisée en minimisant la ϕ -fonctionnelle suivante :

$$\min_{I: \Omega \rightarrow \mathbb{R}} E(I) = \int_{\Omega} \phi(\|\nabla I\|) d\Omega \quad (2.6)$$

où $\phi : \mathbb{R} \rightarrow \mathbb{R}$ est une *fonction croissante*, ayant un impact direct sur le comportement global de la régularisation en pénalisant les fortes normes de gradient. Cette minimisation peut être

effectuée par une descente de gradient, amenant à une *EDP de diffusion*, provenant des équations d'Euler-Lagrange de $E(I)$:

$$\begin{cases} I_{(t=0)} = I_{\text{bruitee}} \\ \frac{\partial I}{\partial t} = \text{div} \left(\frac{\phi'(\|\nabla I\|)}{\|\nabla I\|} \nabla I \right) \end{cases} \quad (2.7)$$

Différents choix de fonctions ϕ amènent à un panel de différentes méthodes de régularisation. On retrouve en particulier la régularisation isotrope (équivalente à une convolution de l'image par une gaussienne), telle qu'introduite par Tikhonov [TIKHON63], mais aussi la méthode classique de Perona-Malik [PERONA90] ou encore de la Variation Totale [RUDIN92]. D'autres méthodes de régularisation agissant sur des images scalaires rentrent aussi dans ce formalisme unificateur des ϕ -fonctionnelles (Fig.2.5).

Méthode	$\phi(s)$	Référence
Tikhonov	s^2	[TIKHON63]
Perona-Malik	$1 - \exp(-s^2/K^2)$	[PERONA90]
Surfaces minimales	$2\sqrt{1+s^2} - 2$	[CHARBO94]
Geman-McClure	$s^2/(1+s^2)$	[GEMAN85]
Variation totale	s	[RUDIN92]
Green	$2\log(\cosh(s))$	[GREEN90]

FIG. 2.5: Différentes fonctions pour la régularisation par ϕ -fonctionnelles.

Une extension naturelle des ϕ -fonctionnelles pour les images \mathbf{I} *multi-canaux* consisterait donc en la minimisation de la fonctionnelle $E(\mathbf{I}) = \int_{\Omega} \phi(\mathcal{N}(\mathbf{I})) d\Omega$, mesurant là encore, une variation globale dans l'image \mathbf{I} , où $\mathcal{N}(\mathbf{I})$ pourrait être l'une des trois mesures de variations locales, telle que définies en section précédente. Mais, comme une image multi-canaux possède deux mesures de variations distinctes λ_+, λ_- (les valeurs propres du tenseur de structure \mathbf{G}), contrairement à la mesure unique $\|\nabla I\|$ des images scalaires, on peut plus généralement s'intéresser à la minimisation de la ψ -fonctionnelle suivante, définie par une fonction $\psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ de deux variables, qui formule une extension plus générique des ϕ -fonctionnelles pour les images multi-canaux :

$$\min_{\mathbf{I}:\Omega \rightarrow \mathbb{R}^n} E(\mathbf{I}) = \int_{\Omega} \psi(\lambda_+, \lambda_-) d\Omega \quad (2.8)$$

Les équations d'Euler-Lagrange de l'Eq.(2.8) débouchent sur cette forme très simple d'EDP de type *Divergence*, qui minimise $E(\mathbf{I})$:

$$\frac{\partial I_i}{\partial t} = \text{div} \left(\left[\frac{\partial \psi}{\partial \lambda_+} \theta_+ \theta_+^T + \frac{\partial \psi}{\partial \lambda_-} \theta_- \theta_-^T \right] \nabla I_i \right) \quad (i = 1..n) \quad (2.9)$$

Là encore, certaines fonctions ψ permettent de retrouver des approches variationnelles de régularisation scalaire ou multi-canaux connues, comme bien sûr l'ensemble du formalisme des ϕ -fonctions [BLOMGR98, TANG00], avec $\psi(\lambda_+, \lambda_-) = \phi(\sqrt{\lambda_+})$ ou encore le formalisme du flot de Beltrami [KIMMEL00, SOCHEN01B, SOCHEN98A, SOCHEN01A], avec $\psi(\lambda_+, \lambda_-) = \sqrt{(1+\lambda_+)(1+\lambda_-)}$. Notons que cette dernière fonctionnelle est proportionnelle à l'*aire de l'image* \mathbf{I} , vue comme une surface 2D dans un espace $(n+2)$ D. On comprend alors mieux

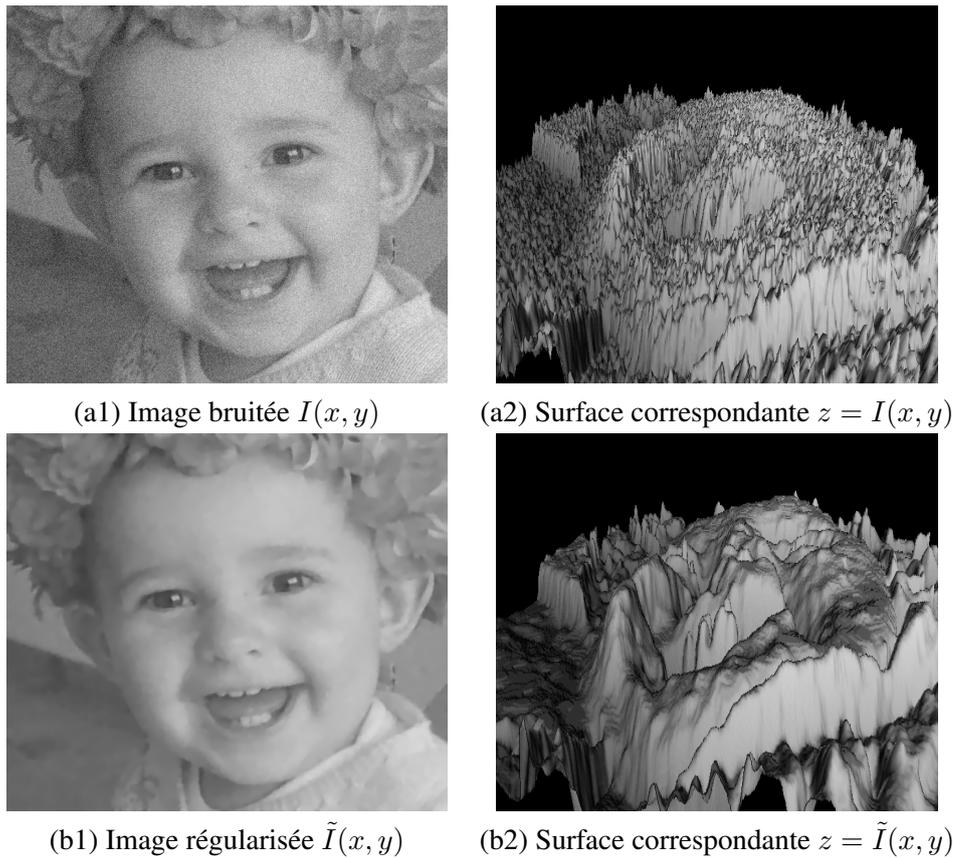


FIG. 2.6: Régularisation d'images, vue comme la minimisation de l'aire d'une surface.

quel rôle peut jouer une telle minimisation de fonctionnelle en terme de lissage/régularisation d'images, ici, en la faisant converger vers une surface d'aire minimale (Fig.2.6).

Malgré cette interprétation géométrique globale intéressante, on peut regretter que ces méthodes variationnelles de régularisation manquent de flexibilité. Elles sont en effet formulées comme des processus de minimisation globale, alors que les propriétés géométriques du lissage que nous souhaitons définir sont au contraire intrinsèquement locales (car le type de dégradation que l'on cherche à corriger est lui-même local). Les EDP minimisantes, obtenues par les équations d'Euler-Lagrange, ne permettent pas de s'adapter finement aux différentes configurations géométriques locales des structures (contours, coins, etc.). Or, cette flexibilité du comportement de lissage est souvent souhaitable, particulièrement lorsque le niveau de bruit dans les images est important.

2.3.2 EDP de type *Divergence*

Les travaux proposés dans [ALVARE92, AUBERT02, KORNP97B, SAPIRO01, WEICKE98]. ont permis l'élaboration d'EDP de diffusion plus génériques et localement plus flexibles. L'idée consiste dans un premier temps à remplacer la diffusivité scalaire $\phi'(\|\nabla I\|)/\|\nabla I\|$ présente dans l'EDP de diffusion Eq.(2.7) par une expression dépendante de caractéristiques locales quelconques des images. Ce faisant, on se rapproche du niveau local désiré du processus de lissage, en acceptant de perdre en contrepartie le niveau d'interprétation global de la régularisation : gé-

néralement, ces EDP élaborées directement ne correspondent plus à des descentes de gradient pour la minimisation de fonctionnelles.

Historiquement, les auteurs de [ALVARE92] ont été les premiers à proposer l'utilisation d'une fonction de diffusivité $g(\|\nabla(I * G_\sigma)\|)$ dépendante, non plus de la norme du gradient $\|\nabla I\|$, mais de la norme du gradient convoluée par une gaussienne $\|\nabla(I * G_\sigma)\|$, pour mesurer des variations géométriques scalaires plus robustes au bruit :

$$\frac{\partial I}{\partial t} = \operatorname{div} (g(\|\nabla(I * G_\sigma)\|) \nabla I)$$

Cette technique assurait que les formulations proposées pour la régularisation soient bien posées. Mais il est apparu que cela permettait aussi de prendre en compte une géométrie de diffusion locale plus cohérente, en faisant intervenir des données provenant de voisinages aux pixels plus étendus (par l'action de la convolution). Par la suite, une généralisation majeure des équations de type *Divergence* pour les images scalaires et multi-canaux a été proposée par Weickert dans [WEICKE97A, WEICKE98]. Il se représente les pixels de l'image comme des concentrations chimiques ou des températures soumises à des lois de diffusion physiques (loi de Fick et équation de continuité). Weickert a proposé ainsi cette équation de type *Divergence*, générique et paramétrée par un champ $\mathbf{D} : \Omega \rightarrow \mathcal{P}(2)$ de tenseurs de diffusion 2×2 :

$$\frac{\partial I_i}{\partial t} = \operatorname{div} (\mathbf{D} \nabla I_i) \quad (i = 1..n) \quad (2.10)$$

Le champ de tenseurs \mathbf{D} définit un *flux de gradient* qui est censé contrôler de manière fine le comportement géométrique local du processus de diffusion de l'Eq.(2.10). Notons que le formalisme des ψ -fonctionnelles décrit en section 2.3.1 devient un cas particulier de l'EDP Eq.(2.10), avec \mathbf{D} défini en chaque point par $\mathbf{D} = \frac{\partial \psi}{\partial \lambda_+} \theta_+ \theta_+^T + \frac{\partial \psi}{\partial \lambda_-} \theta_- \theta_-^T$.

Plus spécifiquement, Weickert a proposé de définir le tenseur de diffusion en chaque point \mathbf{X} de l'image par $\mathbf{D} = \lambda_1 \mathbf{u}_1 \mathbf{u}_1^T + \lambda_2 \mathbf{u}_2 \mathbf{u}_2^T$, c-à-d en choisissant ses deux vecteurs propres $\mathbf{u}_1, \mathbf{u}_2$ et valeurs propres λ_1, λ_2 comme des fonctions des éléments spectraux du tenseur de structure lissé $\mathbf{G}_{\alpha, \sigma}$, telles que :

$$\begin{cases} \mathbf{u}_1 = \theta_+ \\ \mathbf{u}_2 = \theta_- \end{cases} \quad \text{et} \quad \begin{cases} \lambda_1 = \beta \\ \lambda_2 = \begin{cases} \beta & \text{si } \lambda_+ = \lambda_- \\ \beta + (1 - \beta) \exp\left(\frac{-C}{(\lambda_+ - \lambda_-)^2}\right) & \text{sinon} \end{cases} \end{cases} \quad (2.11)$$

($C > 0$ et $\beta \in [0, 1]$ étant des paramètres de la méthode, définis par l'utilisateur).

Dans le cas des images multi-canaux, on peut remarquer que le même champ de tenseurs \mathbf{D} sera utilisé pour orienter la diffusion sur toutes les composantes I_i de l'image, ce qui assure que les I_i soient lissées en considérant une géométrie multi-valuée cohérente qui tient compte de la corrélation éventuelle des données entre tous les canaux de l'image (\mathbf{D} dépend en effet de $\mathbf{G}_{\alpha, \sigma}$), contrairement aux approches traitant chaque composante indépendamment.

Dans ces travaux, Weickert a donc présumé que la forme du tenseur de diffusion en chaque point \mathbf{X} du champ \mathbf{D} est représentative de la géométrie souhaitée du lissage en ce point. Le choix du champ de tenseurs Eq.(2.11) implique donc que :

- Dans les régions de niveaux homogènes, $\lambda_+ \approx \lambda_- \approx 0$ et on obtient donc $\lambda_1 \approx \lambda_2 \approx \beta$, c-à-d $\mathbf{D} \approx \alpha \mathbb{I}_d$ (matrice identité). Le tenseur \mathbf{D} est alors *isotrope* dans ces régions de faibles variations, ce qui est cohérent puisque l'on souhaite effectivement lisser le bruit

dans toutes les directions dans ce genre de zones. L'EDP Eq.(2.10) se simplifie localement en une équation de la chaleur $\frac{\partial I_i}{\partial t} = \Delta I_i$, sur chaque canal I_i , où $\Delta I_i = \frac{\partial^2 I_i}{\partial x^2} + \frac{\partial^2 I_i}{\partial y^2}$ dénote le laplacien de I_i .

- Le long des contours d'images, nous avons par contre $\lambda_+ \gg \lambda_- \gg 0$, et ainsi, $\lambda_2 > \lambda_1 > 0$. Ici, le tenseur de diffusion \mathbf{D} est donc essentiellement *anisotrope* (la valeur de β étant faible), et principalement orienté suivant la direction θ_- des contours de l'image, ce qui semble également cohérent si l'on souhaite éviter le sur-lissage de ces contours.

Néanmoins, il est important de remarquer que les amplitudes et les directions réelles du lissage local effectué par l'EDP Eq.(2.10) *ne sont pas précisément définies par les éléments spectraux* (c-à-d par la forme) du tenseur de diffusion \mathbf{D} localisé au point \mathbf{X} . Il est en effet possible d'obtenir des comportements de lissage contradictoire avec la forme du tenseur, comme cela est illustré par le cas d'école suivant.

Supposons que nous souhaitions lisser de manière purement anisotrope une image scalaire $I : \Omega \rightarrow \mathbb{R}$ partout le long des directions de gradient $\frac{\nabla I}{\|\nabla I\|}$ avec une amplitude de lissage constante de 1. C'est bien sûr un cas peu intéressant en pratique, puisqu'ici, on va favoriser le lissage des discontinuité des images orthogonalement à leur direction, ce qui va entraîner très rapidement un sur-lissage de l'image. Intuitivement, on serait donc tentés de définir \mathbf{D} en chaque point $\mathbf{X} \in \Omega$ par :

$$\forall \mathbf{X} \in \Omega, \quad \mathbf{D} = \left(\frac{\nabla I}{\|\nabla I\|} \right) \left(\frac{\nabla I}{\|\nabla I\|} \right)^T$$

c-à-d un tenseur dont les deux valeurs propres sont 1 et 0, et les vecteurs propres respectifs associés sont $\frac{\nabla I}{\|\nabla I\|}$ et $\frac{\nabla I^\perp}{\|\nabla I^\perp\|}$. Ceci amènerait à la simplification de l'Eq.(2.10) en

$$\frac{\partial I}{\partial t} = \operatorname{div} \left(\frac{1}{\|\nabla I\|^2} \nabla I \nabla I^T \nabla I \right) = \operatorname{div} (\nabla I) = \Delta I$$

Or, l'évolution de cette *équation de la chaleur* est connue pour être similaire à la convolution de l'image I par un filtre gaussien G_σ isotrope, normalisé et d'écart type $\sigma = \sqrt{2t}$ [KOENDE84]. Ainsi, pour un choix particulier de tenseurs \mathbf{D} *purement anisotropes*, on obtient *in-fine* un comportement de régularisation purement *isotrope*, sans orientations préférentielles pour le lissage. Remarquons que si nous avons choisi $\mathbf{D} = \mathbb{I}_d$ (matrice identité), nous aurions obtenu exactement la même équation : ainsi, différents champs \mathbf{D} avec des formes très opposées (isotropes et anisotropes) débouchent finalement sur un même comportement de lissage local isotrope. De fait, la divergence est un opérateur dérivatif. Par conséquent, le lissage défini par l'Eq.(2.10) dépendra toujours implicitement des variations spatiales du champ de tenseurs \mathbf{D} . D'un point de vue pratique, l'EDP Eq.(2.10) de type *Divergence* ne permet pas d'élaborer un comportement de lissage local qui peut être aisément défini point par point.

2.3.3 EDP de type *Laplaciens Orientés*

Les équations de type *Laplaciens Orientés* en $2D$ considèrent qu'un processus de lissage local peut être décomposé en deux équations de la chaleur $1D$, respectivement orientées le long de deux directions orthogonales \mathbf{u}_1 et \mathbf{u}_2 , chacune associée à deux amplitudes de lissage c_1 et c_2 . Les amplitudes et les orientations de lissage sont bien entendu choisies indépendamment en

chaque point, pour s'adapter à la configuration géométrique locale de l'image (Fig.2.7). L'équation résultante s'écrit comme la somme de deux équations de la chaleur mono-dimensionnelles :

$$\frac{\partial \mathbf{I}}{\partial t} = c_1 \mathbf{I}_{\mathbf{u}_1 \mathbf{u}_1} + c_2 \mathbf{I}_{\mathbf{u}_2 \mathbf{u}_2} \quad (2.12)$$

où \mathbf{u}_1 et \mathbf{u}_2 sont des vecteurs orthonormés, et $c_1, c_2 \geq 0$. $\mathbf{I}_{\mathbf{u}_1 \mathbf{u}_1}$ et $\mathbf{I}_{\mathbf{u}_2 \mathbf{u}_2}$ représentent les dérivées secondes de \mathbf{I} dans les directions \mathbf{u}_1 et \mathbf{u}_2 , et chacune de leurs composantes vectorielles est formellement définie par :

$$\forall i = 1..n, \quad I_{\mathbf{u}_i \mathbf{u}_i} = \mathbf{u}_i^T \mathbf{H}_i \mathbf{u}_i \quad \text{et} \quad I_{i_{\mathbf{u}_2 \mathbf{u}_2}} = \mathbf{u}_2^T \mathbf{H}_i \mathbf{u}_2$$

où \mathbf{H}_i est la matrice hessienne de I_i , définie en chaque point $\mathbf{X} \in \Omega$ par

$$\mathbf{H}_i = \begin{pmatrix} I_{i_{xx}} & I_{i_{xy}} \\ I_{i_{xy}} & I_{i_{yy}} \end{pmatrix} = \begin{pmatrix} \frac{\partial^2 I_i}{\partial x^2} & \frac{\partial^2 I_i}{\partial x \partial y} \\ \frac{\partial^2 I_i}{\partial x \partial y} & \frac{\partial^2 I_i}{\partial y^2} \end{pmatrix} \quad (2.13)$$

Ici, le comportement de diffusion est entièrement défini par la donnée des directions de lissage

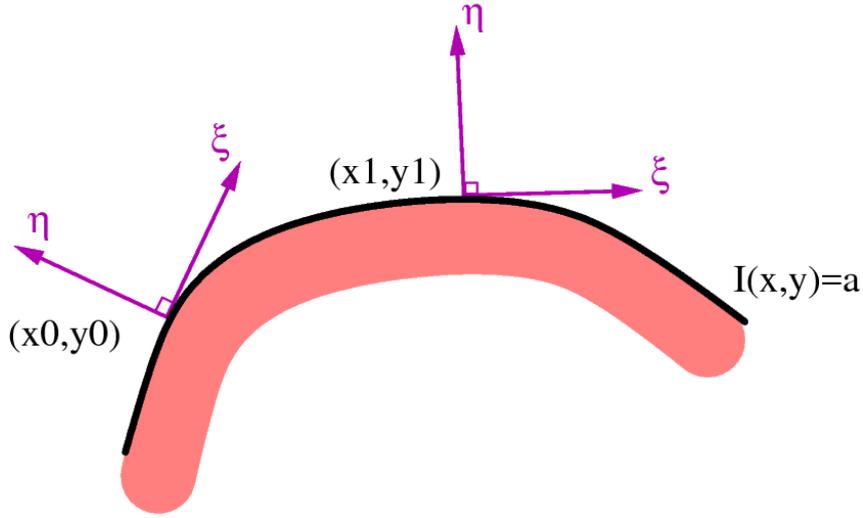


FIG. 2.7: Principe des *Laplaciens Orientés* : deux lissages 1D effectués simultanément le long de directions orthogonales.

$\mathbf{u}_1, \mathbf{u}_2$ et des amplitudes de lissage correspondantes c_1 et c_2 en chaque point \mathbf{X} de l'image. Cette formulation très générale a été initialement proposée dans [KORNPR97A] pour la régularisation d'images scalaires I , avec ce choix des fonctions c_1, c_2 et $\mathbf{u}_1, \mathbf{u}_2$ suivant :

$$\begin{cases} \mathbf{u}_1 = \xi = \frac{\nabla I^\perp}{\|\nabla I\|} \\ \mathbf{u}_2 = \eta = \frac{\nabla I}{\|\nabla I\|} \end{cases} \quad \text{et} \quad \begin{cases} c_1 = 1 \\ c_2 = g(\|\nabla I * G_\sigma\|) \end{cases} \quad (2.14)$$

où $g : \mathbb{R} \rightarrow \mathbb{R}$ est une fonction décroissante vers 0, afin que le lissage effectué orthogonalement aux discontinuités s'atténue sur les régions de contours (c-à-d à forts gradients). Le choix de paramètres par l'Eq.(2.14) permet d'obtenir un lissage anisotrope permanent le long des directions de contours ξ , même dans les zones à très forts gradients, puisque $c_1 = 1$ partout dans l'image. Mais ce formalisme général des *Laplaciens Orientés* permet également de retrouver

de nombreuses autres équations bien connues de la littérature, comme par exemple l'ensemble des ϕ -fonctionnelles, obtenu en choisissant $c_1 = \phi'(\|\nabla I\|)/\|\nabla I\|$, $c_2 = \phi''(\|\nabla I\|)$, $\mathbf{u}_1 = \xi$ et $\mathbf{v}_2 = \eta$, ou encore *le flot de courbure moyenne* $\frac{\partial I}{\partial t} = I_{\xi\xi}$, avec $c_1 = 1$, $c_2 = 0$, $\mathbf{u}_1 = \xi$ et $\mathbf{v}_2 = \eta$ [CARMON98, DERICH95]. Notons que ce dernier type de lissage ne peut pas être formulé par une expression de type *Divergence*.

Ringach et Sapiro [SAPIRO96B] ont été les premiers à proposer une extension du flot de courbure moyenne $I_t = I_{\xi\xi}$ pour les images multi-canaux, en utilisant une formulation *Laplacienne Orientée*. Ils se sont naturellement basés sur les attributs géométriques de Di Zenzo pour incorporer les informations de géométrie multi-valuée dans leur équation de diffusion :

$$\frac{\partial \mathbf{I}}{\partial t} = g(\lambda_+ - \lambda_-) \mathbf{I}_{\theta_- \theta_-} \quad (2.15)$$

où $g : \mathbb{R} \rightarrow \mathbb{R}$ est une fonction positive décroissante, permettant d'éviter le sur-lissage des zones à forts gradients. Ici également, chaque composante I_i est lissée le long *d'une direction commune* θ_- (celle des contours vectoriels) avec *une amplitude commune*. Malgré cette extension astucieuse, plusieurs inconvénients subsistaient pour l'application à la régularisation d'images :

- La mesure de cohérence $\mathcal{N}_- = \sqrt{\lambda_+ - \lambda_-}$ a été utilisée pour déterminer les variations locales multi-valuées dans le but de réduire la diffusion sur les contours de l'image. Ce n'est pas un choix optimal (comme remarqué en section 2.2), puisque sur certaines structures de type "jonction", la mesure \mathcal{N}_- devient faible, entraînant un sur-lissage en ces points.
- Dans les régions localement homogènes (à faibles variations, c-à-d $\mathcal{N}_- \rightarrow 0$), la diffusion est effectuée le long d'une direction unique θ_- , qui sera ici principalement dirigée de manière aléatoire par le bruit présent dans l'image, puisqu'aucune structure géométrique contrastée n'existe pour donner une orientation cohérente. Ce lissage, partout mono-directionnel, entraîne des effets de textures indésirables, particulièrement dans le cas des images multi-canaux où toutes les composantes sont lissées de manière cohérente et ne se mélangent donc pas. Un lissage isotrope serait nettement préférable pour ce type de configuration géométrique.

2.3.4 EDP de type *Trace*

Une formalisation plus simple des EDP de type *Laplaciens Orientés* a été proposée plus tard dans [TSCHUM05]. L'idée repose sur l'utilisation d'un champ de tenseurs de diffusion $\mathbf{T} : \Omega \rightarrow \mathcal{P}(2)$ pour modéliser la géométrie de lissage de l'Eq.(2.12), au lieu de considérer de manière séparée des directions $\mathbf{u}_1, \mathbf{u}_2$ et des amplitudes c_1, c_2 de lissage. L'équation proposée est une réécriture de la précédente EDP Eq.(2.12), pour les images multi-canaux, en utilisant un opérateur *Trace* :

$$\forall i = 1, \dots, n, \quad \frac{\partial I_i}{\partial t} = \text{trace}(\mathbf{T}\mathbf{H}_i) \quad (2.16)$$

où \mathbf{H}_i représente la matrice hessienne de I_i Eq.(2.13) et \mathbf{T} est le champ de tenseurs de diffusion, calculé comme étant $\mathbf{T}_{(\mathbf{x})} = c_1 \mathbf{u}_1 \mathbf{u}_1^T + c_2 \mathbf{v}_2 \mathbf{v}_2^T$. On retrouve ici l'aspect couplé de la régularisation, puisque chaque composante I_i de \mathbf{I} est lissée en considérant le même champ de tenseurs \mathbf{T} .

Mathématiquement, les Eq.(2.12) et Eq.(2.16) sont strictement équivalentes, mais cette dernière fait apparaître plus clairement la séparation entre la géométrie de lissage (définie par le champ de tenseurs \mathbf{T}) du processus de lissage lui-même. L'approche est au final très similaire à

celle de Weickert et de son EDP Eq.(2.10) de type *Divergence*, le problème de la régularisation se ramenant principalement en l'élaboration d'un champ de tenseurs de diffusion adapté à l'application considérée. Mais dans le cas des EDP de type *Trace*, le champ de tenseurs qui définit le comportement de lissage local a une propriété intéressante d'*unicité* : deux champs de tenseurs *différents* vont nécessairement aboutir à deux comportements de lissage *distincts* (l'opérateur *Trace* n'étant pas dérivatif).

L'Eq.(2.16) a en effet une interprétation géométrique simple en terme de *filtrage local* par des noyaux gaussiens orientés. Dans un premier temps, considérons le cas où \mathbf{T} est un champ de tenseurs constant. On peut alors démontrer que la solution formelle de l'EDP Eq.(2.16) est :

$$I_i^{[t]} = I_i^{[t=0]} * G^{(\mathbf{T},t)} \quad (i = 1..n) \quad (2.17)$$

où $*$ dénote l'opérateur de convolution et $G^{(\mathbf{T},t)}$ est le noyau gaussien normalisé et orienté par \mathbf{T} :

$$G^{(\mathbf{T},t)}(\mathbf{X}) = \frac{1}{4\pi t} \exp\left(-\frac{\mathbf{X}^T \mathbf{T}^{-1} \mathbf{X}}{4t}\right) \quad \text{avec } \mathbf{X} = (x \ y)^T \quad (2.18)$$

On obtient alors une version anisotrope de la propriété démontrée par Koenderink [KOENDE84], portant sur le lien entre opérateur de convolution et diffusion isotrope, que l'on peut retrouver ici en choisissant $\mathbf{T} = \mathbb{I}_d$, qui simplifie alors l'EDP Eq.(2.16) en l'équation de la chaleur 2D : $\frac{\partial I_i}{\partial t} = \text{trace}(\mathbf{H}_i) = \Delta I_i$.

La Fig.2.8 illustre cette propriété de géométrie de lissage tensorielle de l'EDP Eq.(2.16) de type *Trace*, avec la représentation de trois noyaux gaussiens $G^{(\mathbf{T},t)}$ respectivement déformés par des tenseurs isotropes et anisotropes \mathbf{T} , et des trois résultats d'évolutions de l'EDP correspondantes, appliquée sur une image couleur. On peut remarquer que les tracés des fonctions gaussiennes $G^{(\mathbf{T},t)}$ représentent justement les tenseurs \mathbf{T} sous forme d'ellipse. Réciproquement, il est net que la forme des tenseurs \mathbf{T} symbolise effectivement la géométrie réelle du lissage effectué par l'EDP Eq.(2.16).



FIG. 2.8: EDP de type *Trace* vue comme une convolution par des noyaux orientés.

Quand \mathbf{T} n'est pas constant (ce qui est généralement le cas), il modélise alors un champ $\Omega \rightarrow \mathbb{P}(2)$ de tenseurs de diffusion variant spatialement. L'EDP Eq.(2.16) devient *non linéaire*

mais peut toujours être vue comme l'application de *masques de convolution locaux* $G^{\mathbf{T},t}(\mathbf{X})$, variant spatialement et temporellement, sur l'image \mathbf{I} . La Fig.2.9 illustre cette propriété, avec trois exemples distincts de champs de tenseurs \mathbf{T} non constants (représentés par des champs d'ellipses), et les résultats d'évolutions de l'EDP Eq.(2.16) correspondants. Comme précédemment, la forme de chaque tenseur \mathbf{T} donne la géométrie précise du processus de lissage local effectué par l'EDP Eq.(2.16) de type *Trace*, et ceci, de manière ponctuelle. Comme l'opérateur *Trace* n'est pas dérivatif, l'interprétation locale du processus de lissage comme une convolution par des masques gaussiens orientés dans un voisinage infinitésimal, reste valide ici.

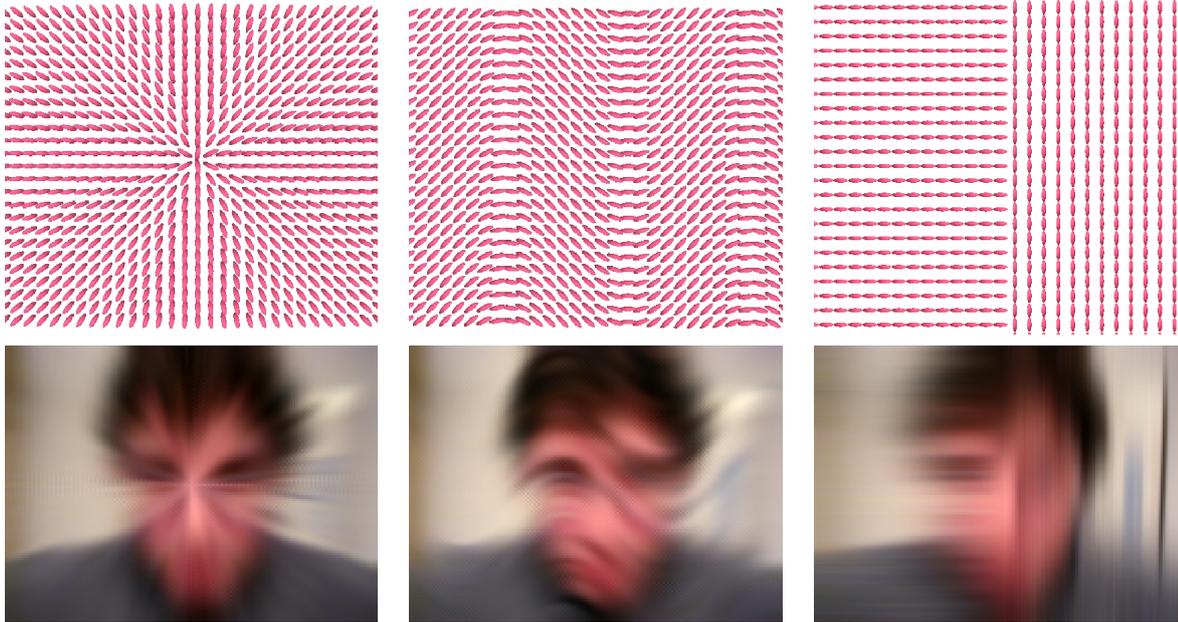


FIG. 2.9: EDP de type *Trace* utilisant des tenseurs de diffusion non-constants.

De la même façon que les tenseurs de structure $\mathbf{G}_{\alpha,\sigma}$ mesurent en chaque point les orientations principales θ_+ , θ_- et les contrastes locaux $\lambda_+ + \lambda_-$ des contours, le champ de tenseurs de diffusion \mathbf{T} représente les orientations préférées du lissage local ainsi que les amplitudes de lissage le long de ces orientations, également en chaque point. Intuitivement, il semble souhaitable de choisir \mathbf{T} en fonction de la géométrie locale de \mathbf{I} , ce qui sous-entend de le faire dépendre des éléments spectraux λ_- , λ_+ et θ_- , θ_+ du tenseur de structure lissé $\mathbf{G}_{\alpha,\sigma}$. Pour le débruitage des images multi-canaux, le choix suivant a été proposée dans [TSCHEUM05] :

$$c_1 = f_{(\lambda_+,\lambda_-)}^- = \frac{1}{(1 + \lambda_+ + \lambda_-)^{p_1}} \quad \text{et} \quad c_2 = f_{(\lambda_+,\lambda_-)}^+ = \frac{1}{(1 + \lambda_+ + \lambda_-)^{p_2}} \quad (2.19)$$

avec $\mathbf{u}_1 = \theta_-$ et $\mathbf{u}_2 = \theta_+$.

Ce choix est raisonnable, car il satisfait un comportement de lissage local naturel pour le débruitage :

- En un point localisé sur un contour de l'image ($\lambda_+ + \lambda_-$ est important), le lissage est effectué principalement le long de la direction du contour θ_- (puisque $f^+ \ll f^-$), avec une amplitude inversement proportionnelle au contraste local du contour.
- En un point localisé dans une région homogène ($\lambda_+ + \lambda_-$ est faible), le lissage est effectué le long de toutes les orientations possibles du plan (lissage isotrope, puisque $f^+ \approx f^-$) et

par conséquent $\mathbf{T} \approx \mathbb{I}_d$ (matrice identité). L'EDP Eq.(2.16) se comporte alors localement comme une équation de la chaleur.

- Les deux paramètres $p_1 < p_2 \in \mathbb{R}$ permettent de définir finement le type d'anisotropie du lissage en fonction de la configuration géométrique locale, et entre ces deux cas extrêmes : si $p_1 \approx p_2$, le lissage est presque partout isotrope, tout en restant localement freiné par les forts contrastes locaux (à privilégier dans le cas d'un bruit important). Si $p_2 \gg p_1$, le lissage est presque partout anisotrope, principalement le long des contours couleurs θ_- , même dans les zones relativement homogènes (à privilégier dans le cas d'un bruit peu important).

L'Eq.(2.16) de type *Trace* est une formulation intéressante qui permet de disjoindre la géométrie de lissage et l'action du lissage, tout en proposant une interprétation géométrique sous forme de filtrage local par des noyaux gaussiens orientés de la façon dont le lissage est effectué. Elle permet de mieux concevoir les liens naturels entre EDP de diffusion et d'autres types de lissage locaux, comme le filtrage Bilatéral [BARASH01, TOMASI98] par exemple. Néanmoins, on peut se demander si ce comportement local *gaussien* du lissage (dont l'amplitude et l'orientation sont données par le tenseur de diffusion \mathbf{T}) est le mieux adapté pour un processus de régularisation. A priori, sur les structures d'images à forte courbure (comme les coins), ce comportement de lissage gaussien n'est pas optimal : lorsque la variation de l'orientation θ_- des contours est importante, un filtrage gaussien local va avoir tendance à *arrondir* ces contours, même lorsque celui-ci est effectué uniquement suivant leur direction θ_- . Cela est dû au fait qu'un masque gaussien orienté tel que l'Eq.(2.18) *n'est pas courbé* lui-même. Ce problème est illustré en Fig.2.10b et Fig.2.11b où de nombreuses itérations de l'EDP Eq.(2.16) ont été appliquées sur des images couleur (synthétique et réelle) et où \mathbf{T} a été défini par l'expression Eq.(2.19). On peut s'apercevoir que les structures à forte courbure dans les images s'arrondissent, résultant en un sur-lissage malvenu des coins du carré dans la Fig.2.10b, où des structures très fines dans l'empreinte digitale de la Fig.2.11b.

Pour prévenir cet effet de sur-lissage, on pourrait imaginer stopper tout simplement l'évolution de l'EDP de diffusion sur les coins (en annulant les tenseurs \mathbf{T} dans ces régions, c-à-d en choisissant $f^- = f^+ = 0$). Mais cela implique la nécessité de détecter explicitement les structures à forte courbure sur des images souvent bruitées ou corrompues, ce qui est en soi un problème mal posé nécessitant des solutions robustes, difficiles ou coûteuses à mettre en œuvre. Le risque devient alors de *sous-lisser* l'image, en laissant du bruit sur les coins et/ou les contours. Le compromis, entre une bonne suppression du bruit, et la préservation des structures à forte courbure, est relativement difficile à trouver, lorsque l'on utilise des EDP Eq.(2.16) de type *Trace*.

Etre capable de prendre en compte la *courbure* des directions de lissage apparaît donc comme une nécessité, et va motiver la suite de notre étude. Par la suite, nous définirons une classe d'EDP de type *Trace* qui permet de lisser une image multi-canaux \mathbf{I} en suivant une géométrie tensorielle \mathbf{T} , *tout en prenant en compte implicitement les courbures des lignes de champ de \mathbf{T}* . A titre de comparaison, des résultats de cette équation de diffusion contrainte sont présentées en Fig.2.10d et Fig.2.11c.

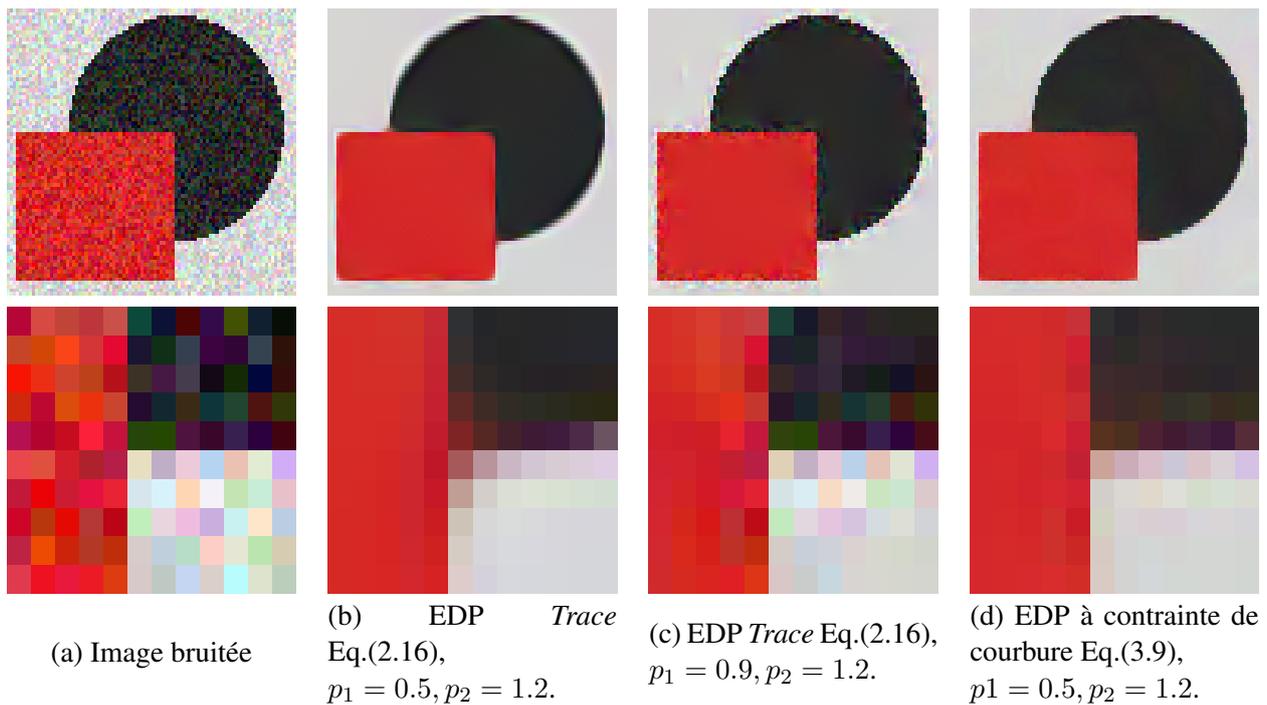


FIG. 2.10: Problèmes de sur-lissage ou sous-lissage rencontrés par l'utilisation d'EDP de type *Trace* sur des structures d'images à forte courbure (détail en deuxième ligne). Le nombre d'itérations a été volontairement exagéré pour accentuer l'effet.

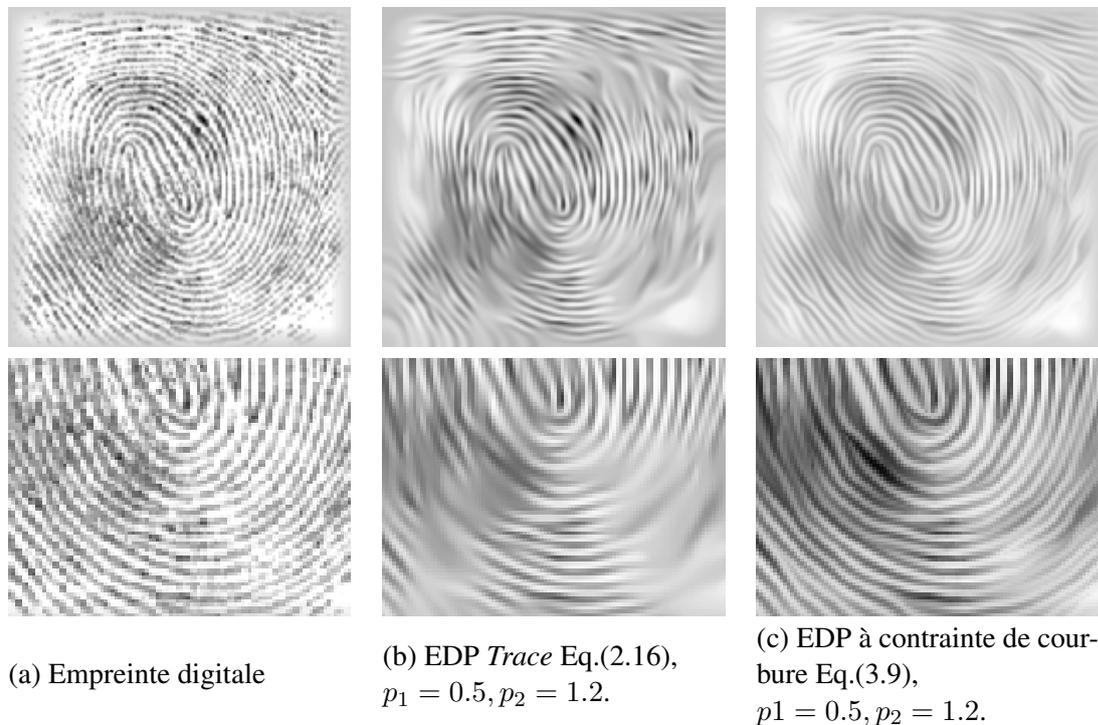


FIG. 2.11: Comparaison entre EDP de type *Trace* et EDP à contrainte de courbure sur une image réelle (détail en deuxième ligne). Le nombre d'itérations a été volontairement exagéré pour accentuer l'effet.

2.4 Conclusion du chapitre

Cet état de l'art des méthodes de lissage anisotropes d'images basées EDP met en lumière deux éléments principaux : d'abord que considérer une modélisation tensorielle de la géométrie locale des images et des processus de diffusion appliqués est plutôt appropriée, puisqu'un tenseur est capable de représenter à la fois des comportements de diffusion *isotropes* et *anisotropes* (contrairement à un vecteur par exemple). Cette propriété de double représentation sera d'ailleurs largement exploitée dans les travaux présentés dans la suite de ce document. D'autre part, on peut remarquer que les formulations EDP classiques de régularisation d'images ne sont pas réellement performantes sur des configurations géométriques difficiles, notamment celles comportant des fortes courbures, avec des problèmes récurrent de sur- ou de sous-lissage des structures géométriques correspondantes. C'est justement à ce point précis que nous nous attaquons dans le chapitre suivant.

EDP de diffusion et préservation des courbures

Sommaire

3.1 Régularisation à préservation de courbure	27
3.2 Un schéma numérique d'implémentation efficace	33
3.3 Applications et résultats	36
3.4 Conclusion du chapitre	37



Nous élaborons ici une variante d'EDP de diffusion anisotrope guidée tenseurs qui se comporte bien lorsque les structures locales des images à régulariser possèdent de fortes courbures (coins, jonctions en T, ...). Nous montrons de plus que notre méthode a un lien direct avec une formulation continue des Convolutions de Lignes Intégrales (LIC), ce qui nous permet de proposer un schéma numérique de lissage très précis (précision sous-pixellique, préservation des structures fines) et en même temps peu coûteux en temps de calcul (peu d'itérations nécessaires). Et ceci, sans perte de généralité sur la classe d'EDP concernée, ce qui nous permet d'appliquer ce schéma pour des applications variées de régularisation d'images couleurs.

Période : 2004–2007.

Publications associées :

- Chapitres de livre : [BC6], [BC5], [BC3], [BC2].
 - Journal : [J6] (IJCV'06).
 - Conférences internationales : [C13] (ECCV-06), [C11] (ICIP'05).
 - Conférence nationale : [NC6] (TAIMA'05).
-

3.1 Régularisation à préservation de courbure

Le but de ces EDP à préservation de courbure est de proposer à la fois une méthode de régularisation générique basée sur une géométrie de lissage tensorielle (comme celle des EDP de type *Divergence* ou *Trace* décrite au chapitre précédent), mais également de se focaliser sur la préservation des structures fines à forte courbure.

3.1.1 Cas d'une direction de lissage unique

Pour illustrer l'idée générale de ces EDP, nous nous limitons dans un premier temps au cas d'une régularisation d'images qui serait dirigée, non pas par un champ de tenseurs \mathbf{T} , mais par un *champ de vecteurs* $\mathbf{w} : \Omega \rightarrow \mathbb{R}^2$. Le lissage local se réalise alors en chaque point le long d'une unique direction $\frac{\mathbf{w}}{\|\mathbf{w}\|}$, avec une amplitude de lissage $\|\mathbf{w}\|$. Nous notons les deux composantes de \mathbf{w} par $\mathbf{w} = (u \ v)^T$.

Nous définissons l'EDP de régularisation à *préservation de courbure* qui lisse \mathbf{I} le long de \mathbf{w} par :

$$\forall i = 1, \dots, n, \quad \frac{\partial I_i}{\partial t} = \text{trace}(\mathbf{w}\mathbf{w}^T \mathbf{H}_i) + \nabla I_i^T \mathbf{J}_w \mathbf{w} \quad (3.1)$$

où \mathbf{J}_w est la matrice Jacobienne de \mathbf{w} , et \mathbf{H}_i est la matrice hessienne de I_i .

$$\mathbf{J}_w = \begin{pmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{pmatrix} \quad \text{et} \quad \mathbf{H}_i = \begin{pmatrix} \frac{\partial^2 I_i}{\partial x^2} & \frac{\partial^2 I_i}{\partial x \partial y} \\ \frac{\partial^2 I_i}{\partial x \partial y} & \frac{\partial^2 I_i}{\partial y^2} \end{pmatrix}$$

L'EDP Eq.(3.1) possède simplement un terme additionnel $\nabla I_i^T \mathbf{J}_w \mathbf{w}$ par rapport à l'équation de type *Trace* vue précédemment, qui ici, lisserait l'image \mathbf{I} le long de \mathbf{w} par des filtrages gaussiens localement orientés par \mathbf{w} (section 2.3.4). Ce terme supplémentaire dépend de la variation du champ de vecteurs \mathbf{w} (et donc implicitement de la courbure de ses lignes de champ). Considérons maintenant la courbe $\mathcal{C}_{(a)}^{\mathbf{X}}$ définissant une *ligne de champ* de \mathbf{w} , partant d'un point \mathbf{X} et paramétré par $a \in \mathbb{R}$ (Fig.3.1) :

$$\begin{cases} \mathcal{C}_{(0)}^{\mathbf{X}} = \mathbf{X} \\ \frac{\partial \mathcal{C}_{(a)}^{\mathbf{X}}}{\partial a} = \mathbf{w}(\mathcal{C}_{(a)}^{\mathbf{X}}) \end{cases} \quad (3.2)$$

Nous appelons \mathcal{F} la famille des lignes de champ de \mathbf{w} .

Le développement de Taylor au 2^{ème} ordre de $\mathcal{C}_{(a)}^{\mathbf{X}}$ autour de $a = 0$ donne :

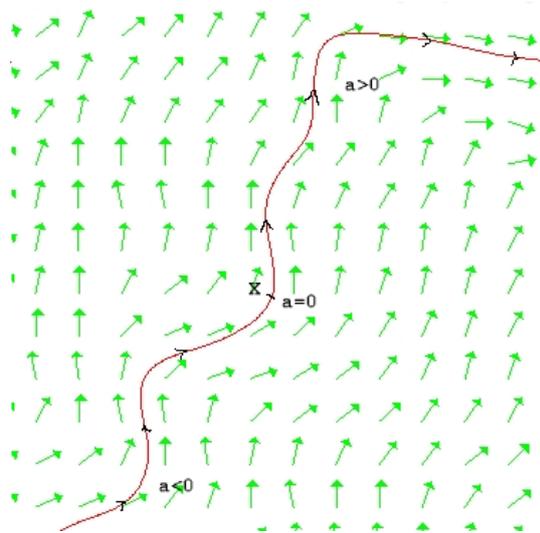
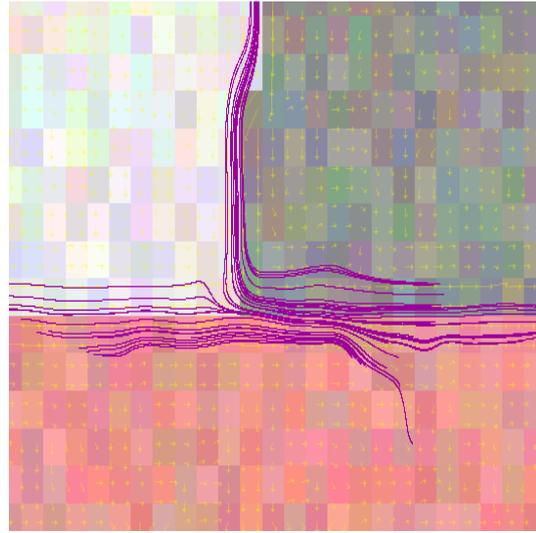
$$\begin{aligned} \mathcal{C}_{(h)}^{\mathbf{X}} &= \mathcal{C}_{(0)}^{\mathbf{X}} + h \frac{\partial \mathcal{C}_{(a)}^{\mathbf{X}}}{\partial a} \Big|_{a=0} + \frac{h^2}{2} \frac{\partial^2 \mathcal{C}_{(a)}^{\mathbf{X}}}{\partial a^2} \Big|_{a=0} + O(h^3) \\ &= \mathbf{X} + h \mathbf{w}_{(\mathbf{X})} + \frac{h^2}{2} \mathbf{J}_{\mathbf{w}_{(\mathbf{X})}} \mathbf{w}_{(\mathbf{X})} + O(h^3) \end{aligned}$$

avec $h \rightarrow 0$, et $O(h^n) = h^n \epsilon_n$. Nous pouvons alors en déduire le développement de Taylor au 2^{ème} ordre de $I_i(\mathcal{C}_{(a)}^{\mathbf{X}})$ autour de $a = 0$, qui donne une information sur les variations des valeurs de l'image au voisinage de \mathbf{X} lorsque l'on se déplace le long de la ligne de champ $\mathcal{C}^{\mathbf{X}}$:

$$I_i(\mathcal{C}_{(h)}^{\mathbf{X}}) = I_i(\mathbf{X}) + h \nabla I_i^T(\mathbf{X}) (\mathbf{w}_{(\mathbf{X})}) + \frac{h^2}{2} \mathbf{J}_{\mathbf{w}_{(\mathbf{X})}} \mathbf{w}_{(\mathbf{X})} + \frac{h^2}{2} \text{trace}(\mathbf{w}_{(\mathbf{X})} \mathbf{w}_{(\mathbf{X})}^T \mathbf{H}_i(\mathbf{X})) + O(h^3)$$

A noter que $\text{trace}(\mathbf{w}_{(\mathbf{X})} \mathbf{w}_{(\mathbf{X})}^T \mathbf{H}_i(\mathbf{X})) = \frac{\partial^2 I_i}{\partial \mathbf{w}^2}$ correspond à la dérivée seconde directionnelle de I_i le long de \mathbf{w} . La dérivée seconde directionnelle de la fonction $a \rightarrow I_i(\mathcal{C}_{(a)}^{\mathbf{X}})$ en $a = 0$ s'écrit alors :

$$\begin{aligned} \frac{\partial^2 I_i(\mathcal{C}_{(a)}^{\mathbf{X}})}{\partial a^2} \Big|_{a=0} &= \lim_{h \rightarrow 0} \frac{1}{h^2} [I_i(\mathcal{C}_{(h)}^{\mathbf{X}}) + I_i(\mathcal{C}_{(-h)}^{\mathbf{X}}) - 2I_i(\mathcal{C}_{(0)}^{\mathbf{X}})] \\ &= \text{trace}(\mathbf{w}_{(\mathbf{X})} \mathbf{w}_{(\mathbf{X})}^T \mathbf{H}_i(\mathbf{X})) + \nabla I_i^T \mathbf{J}_{\mathbf{w}_{(\mathbf{X})}} \mathbf{w}_{(\mathbf{X})} \end{aligned} \quad (3.3)$$

(a) Ligne de champ d'un champ de vecteur w .(b) Exemple de lignes de champ quand w est choisi comme le 2^{ème} vecteur propre θ_- du tenseur de structure lissé $G_{\alpha,\sigma}$ pour une image couleur I (ici, un bloc représente un pixel).FIG. 3.1: Lignes de champ C^X de champs de vecteurs $w : \Omega \rightarrow \mathbb{R}^2$.

qui se trouve être exactement le terme de vitesse de l'EDP Eq.(3.1) de diffusion à préservation de courbure. Plus globalement, l'EDP Eq.(3.1) peut être vue pour chaque ligne de champ \mathcal{F} :

$$\forall C \in \mathcal{F}, \quad \forall a \in \mathbb{R}, \quad \frac{\partial I_i(C(a))}{\partial t} = \frac{\partial^2 I_i(C(a))}{\partial a^2} \quad (3.4)$$

Nous pouvons alors reconnaître dans l'Eq.(3.4) une *équation de la chaleur mono-dimensionnelle sur la ligne de champ C* qui est donc très différente d'une équation de la chaleur *orientée* par w , comme dans la formulation $\frac{\partial I_i}{\partial t} = \frac{\partial^2 I_i}{\partial w^2}$ car ici, les courbures des lignes de champ de w vont être implicitement prises en compte. En particulier, cette EDP a la propriété de s'annuler quand les valeurs de l'image sont parfaitement constantes le long des lignes de champ C de w , et ceci, quelques soient leurs courbures. Dans un contexte de régularisation, définir un champ w qui est partout tangent aux structures importantes de l'image va permettre la préservation de celles-ci lors du lissage, *même si elles ont une forte courbure* (comme sur les coins par exemple). Cette courbure ne peut pas être prise en compte aussi naturellement avec les EDP de type *Divergence* ou *Trace*. Cette propriété de préservation de la courbure de l'Eq.(3.1) est illustrée en Fig.2.10d et Fig.2.11b. Nous montrons de plus dans la section suivante, qu'une approximation rapide de la solution de l'Eq.(3.1) au temps t peut être estimée par une approche analytique bien posée.

3.1.2 Analogie avec les Convolutions de Lignes Intégrales

Les Convolutions de Lignes Intégrales (*LIC*) ont été initialement proposées dans [CABRAL93] comme une technique de rendu d'images texturées pour la représentation d'un champ de vecteurs quelconque $w : \Omega \rightarrow \mathbb{R}^2$. L'idée initiale, exprimée par une formulation discrète, consiste à lisser une image I_{bruitee} de bruit pur, en moyennant les valeurs de pixels le long de courbes

intégrales (c-à-d des lignes de champ) de w . En formulation continue, ce moyennage s'écrit comme :

$$\forall \mathbf{X} \in \Omega, \quad \mathbf{I}_{(\mathbf{X})}^{LIC} = \frac{1}{N} \int_{-\infty}^{+\infty} f(p) \mathbf{I}_{\text{bruitee}}(\mathcal{C}_{(p)}^{\mathbf{X}}) dp \quad (3.5)$$

où $f : \mathbb{R} \rightarrow \mathbb{R}$ est une fonction paire (décroissante vers 0 sur \mathbb{R}^+) et $\mathcal{C}^{\mathbf{X}}$ défini comme précédemment, la *ligne de champ* de w partant du point \mathbf{X} . Le facteur de normalisation $N = \int_{-\infty}^{+\infty} f(p) dp$ moyenne les valeurs des pixels sur $\mathcal{C}^{\mathbf{X}}$.

Comme nous l'avons remarqué en section précédente, l'EDP Eq.(3.1) à préservation de courbure peut être vue comme une équation de la chaleur mono-dimensionnelle Eq.(3.4) sur la ligne de champ $\mathcal{C}^{\mathbf{X}} \in \mathcal{F}$. En utilisant la substitution de variable $\mathbf{L}_{(a)} = \mathbf{I}(\mathcal{C}_{(a)}^{\mathbf{X}})$, l'Eq.(3.4) peut se réécrire comme $\frac{\partial \mathbf{L}}{\partial t}(a) = \mathbf{L}_{(a)}''$. La solution $\mathbf{L}^{[t]}$ au temps t de cette équation de la chaleur est alors connue pour être la convolution 1D de $\mathbf{L}^{[t=0]}$ par le noyau gaussien normalisé $G_{\sqrt{2t}}$ [DERICH95, KOENDE84], d'où :

$$\mathbf{L}_{(a)}^{[t]} = \int_{-\infty}^{+\infty} \mathbf{L}_{(p)}^{[t=0]} G_{\sqrt{2t}(a-p)} dp \quad \text{avec} \quad G_{\sqrt{2t}(p)} = \frac{1}{\sqrt{4\pi t}} \exp\left(-\frac{p^2}{4t}\right) \quad (3.6)$$

Substituer \mathbf{L} par \mathbf{I} dans l'Eq.(3.6) avec $a = 0$, tout en considérant que $\mathcal{C}_{(0)}^{\mathbf{X}} = \mathbf{X}$ et $G_{t(-p)} = G_{t(p)}$ donne :

$$\forall \mathbf{X} \in \Omega, \quad \mathbf{I}_{(\mathbf{X})}^{[t]} = \int_{-\infty}^{+\infty} \mathbf{I}^{[t=0]}(\mathcal{C}_{(p)}^{\mathbf{X}}) G_{\sqrt{2t}(p)} dp \quad (3.7)$$

On retrouve dans l'Eq.(3.7) une forme particulière de l'équation continue des LIC Eq.(3.5), avec une pondération gaussienne $f = G_{\sqrt{2t}}$, et un facteur de normalisation

$$N = \int_{-\infty}^{+\infty} G_{\sqrt{2t}(p)} dp = 1$$

D'un point de vue filtrage local, l'évolution de l'EDP Eq.(3.1) à préservation de courbure peut donc être interprétée comme l'application de convolutions locales infinitésimales par des masques gaussiens locaux 1D le long des lignes de champ \mathcal{C} de w . Ce type de lissage anisotrope amène *in-fine* un filtrage gaussien *courbé*, plutôt que simplement orienté.

L'application de cette équation sur une image multi-canaux \mathbf{I} , en choisissant w comme le champ de vecteurs propres secondaires du tenseur de structure lissé $\mathbf{G}_{\alpha, \sigma}$ (c-à-d. la direction des contours) permet un lissage anisotrope de \mathbf{I} avec préservation des contours, même si ceux-ci sont très courbés. Ceci est illustré en Fig.3.1b, où quelques lignes de champ $\mathcal{C}^{\mathbf{X}}$ sont représentées, autour d'une structure de type "jonction en T". Notez comme les lignes de champ tournent autour de la jonction, avec une précision sous-pixellique. Le calcul des lignes de champ se fait de manière robuste par des schémas classiques de type Runge-Kutta [PRESS92]. Il faut noter cependant que l'Eq.(3.7) est une solution analytique de l'Eq.(3.1) seulement lorsque w *ne varie pas au cours du temps*, ce qui n'est pas le cas dans le cas général des EDP non-linéaires, puisque la géométrie du lissage souhaitée est ré-évaluée à chaque itération, ce qui introduit une variation temporelle de celle-ci. Pour la prendre en compte, il est donc nécessaire d'effectuer explicitement plusieurs itérations consécutives du schéma LIC Eq.(3.7), avec un champ de vecteur w réévalué itérativement. C'est également ce qui est réalisé par les schémas d'Euler explicites plus classiques de résolution d'EDP de diffusion qui supposent que la géométrie de lissage peut être considérée comme constante entre deux instants successifs $\mathbf{I}^{[t]}$ et $\mathbf{I}^{[t+dt]}$, si dt est suffisamment petit.

3.1.3 Une formulation entre *Traces* et *Divergences*

Nous illustrons ici comment l'EDP à préservation de courbure mono-directionnelle Eq.(3.1) se situe par rapport aux expressions de type *Trace* Eq.(2.16) et *Divergence* Eq.(2.10) vues précédemment. Il suffit de développer l'EDP *Divergence* Eq.(2.10) pour le choix particulier du tenseur de diffusion $\mathbf{D} = \mathbf{w} \mathbf{w}^T$:

$$\begin{aligned}
\operatorname{div}(\mathbf{w} \mathbf{w}^T \nabla I_i) &= \operatorname{div} \begin{pmatrix} u^2 \frac{\partial I_i}{\partial x} + uv \frac{\partial I_i}{\partial y} \\ uv \frac{\partial I_i}{\partial x} + v^2 \frac{\partial I_i}{\partial y} \end{pmatrix} \\
&= \left(u^2 \frac{\partial^2 I_i}{\partial x^2} + 2uv \frac{\partial^2 I_i}{\partial x \partial y} + v^2 \frac{\partial^2 I_i}{\partial y^2} \right) + \nabla I_i^T \begin{pmatrix} 2u \frac{\partial u}{\partial x} + u \frac{\partial v}{\partial y} + v \frac{\partial u}{\partial y} \\ 2v \frac{\partial v}{\partial y} + u \frac{\partial v}{\partial x} + v \frac{\partial u}{\partial x} \end{pmatrix} \\
&= \operatorname{trace}(\mathbf{w} \mathbf{w}^T \mathbf{H}_i) + \nabla I_i^T \left[\begin{pmatrix} u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \\ u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \end{pmatrix} + \begin{pmatrix} u \frac{\partial u}{\partial x} + u \frac{\partial v}{\partial y} \\ v \frac{\partial u}{\partial x} + v \frac{\partial v}{\partial y} \end{pmatrix} \right] \\
&= \operatorname{trace}(\mathbf{w} \mathbf{w}^T \mathbf{H}_i) + \nabla I_i^T \mathbf{J}_{\mathbf{w}} \mathbf{w} + \operatorname{div}(\mathbf{w}) \nabla I_i^T \mathbf{w}
\end{aligned}$$

On peut ainsi reconnaître trois termes différents :

- Le premier terme correspond à l'EDP de type *Trace* Eq.(2.16), qui lisse localement \mathbf{I} parallèlement à \mathbf{w} , en utilisant des filtrages gaussiens orientés par \mathbf{w} .
- La somme des deux premiers termes correspond à l'EDP proposée de régularisation à *préservation de courbure* Eq.(3.1), qui lisse localement \mathbf{I} le long de \mathbf{w} en tenant compte de la courbure des lignes de champ \mathcal{C} de \mathbf{w} .
- La somme de tous les termes correspond à l'EDP de type *Divergence* Eq.(2.10) qui effectue une diffusion locale de \mathbf{I} suivant \mathbf{w} (qui est le vecteur propre principal du tenseur $\mathbf{w} \mathbf{w}^T$). C'est ce dernier terme $\operatorname{div}(\mathbf{w}) \nabla I_i^T \mathbf{w}$ qui est principalement responsable des perturbations possibles sur la géométrie effective du lissage effectué par l'équation *Divergence*, comme mentionné en section 2.3.2. Ce terme est en réalité pénalisant pour le lissage d'images.

Il est amusant d'observer que l'EDP à préservation de courbure Eq.(3.1) est donc "mathématiquement" positionnée entre les formulations *Trace* Eq.(2.16) et *Divergence* Eq.(2.10), et permet à la fois de respecter une géométrie pré-définie de lissage \mathbf{w} définie point à point, tout en prenant en compte explicitement la courbure de cette géométrie de lissage.

Notons également que nous pouvons aussi écrire l'EDP à préservation de courbure Eq.(3.1) comme une EDP de type *Divergence*, moins un terme de contrainte :

$$\operatorname{trace}(\mathbf{w} \mathbf{w}^T \mathbf{H}_i) + \nabla I_i^T \mathbf{J}_{\mathbf{w}} \mathbf{w} = \operatorname{div}(\mathbf{w} \mathbf{w}^T \nabla I_i) - \operatorname{div}(\mathbf{w}) \nabla I_i^T \mathbf{w}$$

Deux cas singuliers de directions \mathbf{w} existent dans le cas des images scalaires ($n = 1$) :

- Quand $\mathbf{w} = \frac{\nabla I^\perp}{\|\nabla I\|}$ (direction des isophotes), alors $\nabla I^T \mathbf{J}_{\mathbf{w}} \mathbf{w} = -I_{\mathbf{w}\mathbf{w}}$, ce qui annule alors la vitesse d'évolution de l'équation à préservation de courbure Eq.(3.1), en contrebalançant le terme uniquement basé *Trace* (qui est ici la *vitesse de courbure moyenne*). Aucun

lissage ne sera donc effectué dans ce cas, ce qui est naturel car les valeurs des pixels situés sur les isophotes sont constantes par définition, et moyenner ces valeurs n'aura pas d'incidence sur l'aspect final de l'image. Notons que la vitesse d'évolution de l'EDP de type *Divergence* Eq.(2.10) correspondante $\text{div}(\mathbf{w}\mathbf{w}^T \nabla I_i)$ s'annule également dans ce cas.

- Quand $\mathbf{w} = \frac{\nabla I}{\|\nabla I\|}$ (direction du gradient), alors $\nabla I^T \mathbf{J}_w \mathbf{w} = 0$, et la vitesse d'évolution de l'EDP à préservation de courbure Eq.(3.1) se simplifie en $I_{\mathbf{w}\mathbf{w}}$, qui correspond effectivement à un lissage de l'image le long de la direction du gradient (idem pour l'EDP de type *Trace* Eq.(2.16) dans ce cas). Ici par contre, la vitesse d'évolution de l'EDP de type *Divergence* Eq.(2.10) correspondante devient $\text{div}(\nabla I) = \Delta I$, qui amène à un type de lissage *isotrope* de l'image, plutôt qu'au lissage attendu (partout anisotrope).

Ces cas particuliers permettent de mettre en lumière les quelques différences, relativement subtiles, entre les formulations de type *Trace*, *Divergence*, et à préservation de courbure. Notons finalement, que dans le cas où \mathbf{w} est un champ de vecteur à divergence nulle (c-à-d. $\text{div}(\mathbf{w}) = 0$), alors les EDP *Divergence* Eq.(2.10) et à préservation de courbure Eq.(3.1) sont strictement équivalentes, mais c'est un cas qu'il est peut probable de rencontrer en pratique.

3.1.4 Extension au lissage guidé tenseurs (multi-directionnel)

L'EDP Eq.(3.1) proposée de lissage mono-directionnelle à préservation de courbure peut être étendue pour être dirigée par une géométrie de lissage de type tensorielle $\mathbf{T} : \Omega \rightarrow \mathbb{P}(2)$, plutôt que simplement vectorielle $\mathbf{w} : \Omega \rightarrow \mathbb{R}^2$. Cette extension est primordiale, puisqu'un tenseur de diffusion est capable de décrire des comportements de lissage bien plus flexibles et complexes qu'avec une direction unique : il peut modéliser à la fois des lissages de type *anisotrope* et *isotrope*. Le problème n'est pas trivial, puisque les notions de courbure et de lignes de champ pour des champs de tenseurs \mathbf{T} ne sont pas aussi naturelles à définir que pour des champs de vecteurs \mathbf{w} .

Pour ce faire, le processus de lissage à géométrie tensorielle est décomposé comme un ensemble de processus élémentaires à géométrie mono-directionnelle (vectorielle), le long de toutes les orientations du plan. En remarquant dans un premier temps que

$$\int_{\alpha=0}^{\pi} a_{\alpha} a_{\alpha}^T d\alpha = \frac{\pi}{2} \mathbb{I}_d \quad \text{où} \quad a_{\alpha} = \begin{pmatrix} \cos \alpha \\ \sin \alpha \end{pmatrix}$$

on en déduit une décomposition de tout tenseur \mathbf{T} de dimension 2 sous la forme :

$$\mathbf{T} = \frac{2}{\pi} \sqrt{\mathbf{T}} \left(\int_{\alpha=0}^{\pi} a_{\alpha} a_{\alpha}^T d\alpha \right) \sqrt{\mathbf{T}}$$

où $\sqrt{\mathbf{T}} = \sqrt{f^+} \mathbf{u}\mathbf{u}^T + \sqrt{f^-} \mathbf{v}\mathbf{v}^T$ dénote la racine carrée (au sens matriciel) du tenseur $\mathbf{T} = f^+ \mathbf{u}\mathbf{u}^T + f^- \mathbf{v}\mathbf{v}^T$. On peut vérifier que $(\sqrt{\mathbf{T}})^2 = \mathbf{T}$ et $(\sqrt{\mathbf{T}})^T = \sqrt{\mathbf{T}}$. Le tenseur \mathbf{T} peut alors s'écrire comme :

$$\mathbf{T} = \frac{2}{\pi} \int_{\alpha=0}^{\pi} (\sqrt{\mathbf{T}} a_{\alpha})(\sqrt{\mathbf{T}} a_{\alpha})^T d\alpha \quad (3.8)$$

Intuitivement, le tenseur \mathbf{T} a été divisé en une somme de *tenseurs élémentaires* $(\sqrt{\mathbf{T}} a_{\alpha})(\sqrt{\mathbf{T}} a_{\alpha})^T$, chacun étant purement anisotrope (une seule valeur propre non nulle) et orienté le long d'une direction unique $\sqrt{\mathbf{T}} a_{\alpha} \in \mathbb{R}^2$.

L'Eq.(3.8) suggère ainsi que toute EDP de diffusion à géométrie tensorielle peut se décomposer en une somme de processus de lissage à géométrie vectorielle, de façon à respecter la géométrie tensorielle globale \mathbf{T} que l'on s'est fixée. En particulier :

- Lorsque $\mathbf{T} = \mathbb{I}_d$ (matrice identité), le tenseur est isotrope et $\forall \alpha \in [0, \pi]$, $\sqrt{\mathbf{T}}a_\alpha = a_\alpha$. Le lissage résultant sera donc bien effectué dans toutes les directions a_α du plan, avec la même amplitude de lissage dans chaque direction.
- Si $\mathbf{T} = \mathbf{u}\mathbf{u}^T$ (où $\mathbf{u} \in S^1$ est une direction quelconque), le tenseur est purement anisotrope et $\forall \alpha \in [0, \pi]$, $\sqrt{\mathbf{T}}a_\alpha = (\mathbf{u}^T a_\alpha)\mathbf{u}$. Le lissage résultant sera alors vraiment effectué uniquement le long de la direction principale \mathbf{u} du tenseur \mathbf{T} , et ce, pour tous les angles α du plan.

Aussi, en utilisant la décomposition Eq.(3.8) et en considérant que chaque lissage mono-directionnel devrait être effectué préférentiellement par une approche préservant les courbures telle que l'Eq.(3.1), on en déduit l'EDP de régularisation à préservation de courbure suivante, agissant sur une image multi-canaux $\mathbf{I} : \Omega \rightarrow \mathbb{R}^n$ en suivant une géométrie de lissage *tensorielle* \mathbf{T} :

$$\frac{\partial I_i}{\partial t} = \frac{2}{\pi} \int_{\alpha=0}^{\pi} \text{trace} \left((\sqrt{\mathbf{T}}a_\alpha)(\sqrt{\mathbf{T}}a_\alpha)^T \mathbf{H}_i \right) + \nabla I_i^T \mathbf{J}_{\sqrt{\mathbf{T}}a_\alpha} \sqrt{\mathbf{T}}a_\alpha d\alpha$$

qui peut être simplifiée en :

$$\frac{\partial I_i}{\partial t} = \text{trace}(\mathbf{T}\mathbf{H}_i) + \frac{2}{\pi} \nabla I_i^T \int_{\alpha=0}^{\pi} \mathbf{J}_{\sqrt{\mathbf{T}}a_\alpha} \sqrt{\mathbf{T}}a_\alpha d\alpha \quad (3.9)$$

où $a_\alpha = (\cos \alpha \ \sin \alpha)^T$, et $\mathbf{J}_{\sqrt{\mathbf{T}}a_\alpha}$ est la matrice jacobienne du champ de vecteurs $\Omega \rightarrow \sqrt{\mathbf{T}}a_\alpha$. On retrouve, comme pour le cas mono-directionnel, une EDP Eq.(2.16) de type *Trace* munie d'un terme additionnel permettant de prendre en compte une contrainte de courbure spécifique sur les lignes de champ inhérentes à la géométrie tensorielle \mathbf{T} .

3.2 Un schéma numérique d'implémentation efficace

L'implémentation efficace de la méthode de régularisation Eq.(3.9) peut se réaliser en s'inspirant de l'interprétation géométrique en terme de *LIC* des EDP à préservation de courbure mono-directionnelles, présentée en section 3.1.2. L'évolution de l'EDP Eq.(3.9) peut se faire de manière explicite, par le schéma d'Euler suivant :

$$\begin{cases} \mathbf{I}^{[t=0]} &= \mathbf{I}_{\text{bruitee}} \\ \mathbf{I}^{[t+dt]} &= \mathbf{I}^{[t]} + \frac{2dt}{N} \left(\sum_{k=0}^{N-1} \mathcal{R}(\sqrt{\mathbf{T}}a_\alpha) \right) \end{cases}$$

où $\alpha = k\pi/N$ est un angle discretisé dans $[0, \pi]$ parcourant le demi-plan, dt est le pas de temps usuel de la discrétisation temporelle du flot EDP, et $\mathcal{R}(\mathbf{w})$ représente une discrétisation de la vitesse d'évolution de l'EDP Eq.(3.1) pour le lissage mono-directionnelle préservant les courbures le long du champ de vecteur \mathbf{w} . En réécrivant cette expression comme

$$\mathbf{I}^{[t+dt]} = \frac{1}{N} \left(\sum_{k=0}^{N-1} \mathbf{I}^{[t]} + 2dt \mathcal{R}(\sqrt{\mathbf{T}}a_\alpha) \right)$$

on peut la considérer comme le moyennage de plusieurs processus *LIC* pondérés par des fonctions gaussiennes 1D, le long de l'ensemble des lignes de champ pour les champs de vecteurs $\sqrt{\mathbf{T}}a_\alpha$:

$$\mathbf{I}^{[t+dt]} = \frac{1}{N} \left(\sum_{k=0}^{N-1} \mathbf{I}_{LIC(\sqrt{\mathbf{T}}a_\alpha)}^{[t]} \right) ,$$

où l'écart type de chaque gaussienne est $\sqrt{2dt}$. La difficulté d'implémentation se concentre donc ici sur le calcul d'un *LIC*, qui nécessite un schéma d'intégration des lignes de champ. Pour ce genre de tâche, les méthodes d'intégration numériques de Runge-Kutta [PRESS92] sont dans notre cas suffisamment robustes. A noter que des implémentations de *LIC* plus astucieuses et rapides ont été proposées [STALLI95], mais elles présupposent l'utilisation de fonctions portes de pondération, alors qu'on doit ici pondérer par des fonctions gaussiennes.

Le schéma numérique d'évolution de l'EDP Eq.(3.9) à préservation de courbure peut alors se résumer, pour chaque itération, en :

1. Calcul du champ de tenseurs de structure lissé $\mathbf{G}_{\alpha,\sigma}$ à partir de $\mathbf{I}^{[t]}$:

$$\mathbf{G}_{\alpha,\sigma} = G_\alpha * \sum_{i=1}^n \begin{pmatrix} \left(\frac{\partial I_{i\alpha}^{[t]}}{\partial x} \right)^2 & \left(\frac{\partial I_{i\alpha}^{[t]}}{\partial x} \right) \left(\frac{\partial I_{i\alpha}^{[t]}}{\partial y} \right) \\ \left(\frac{\partial I_{i\alpha}^{[t]}}{\partial x} \right) \left(\frac{\partial I_{i\alpha}^{[t]}}{\partial y} \right) & \left(\frac{\partial I_{i\alpha}^{[t]}}{\partial y} \right)^2 \end{pmatrix}$$

Les paramètres α, σ de régularité de la géométrie tensorielle de lissage sont généralement estimés en fonction du niveau de bruit présent dans l'image.

2. Calcul des valeurs propres λ_+, λ_- et vecteurs propres θ_+, θ_- de $\mathbf{G}_{\alpha,\sigma}$.
3. Calcul du champ de tenseur \mathbf{T} donnant la géométrie désirée du lissage, à partir des éléments spectraux de $\mathbf{G}_{\alpha,\sigma}$, par exemple :

$$\mathbf{T} = \frac{1}{(1 + \lambda_+ + \lambda_-)^{p_1}} \theta_- \theta_-^T + \frac{1}{(1 + \lambda_+ + \lambda_-)^{p_2}} \theta_+ \theta_+^T$$

4. Pour chaque angle $\alpha \in [0, \pi]$, discrétisé par un pas angulaire fixe d_α :
 - 4.1) Calcul du champ de vecteurs $\mathbf{w} = \sqrt{\mathbf{T}} a_\alpha$.
 - 4.2) Calcul d'une Convolution de Ligne Intégrale de $\mathbf{I}^{[t]}$ le long des lignes de champ $\mathcal{C} \in \mathcal{F}$ de \mathbf{w} .
5. Moyennage de toutes les images de *LIC* calculées à l'étape 4.

Les paramètres principaux de cet algorithme sont donc p_1, p_2, σ, dt et nb , le nombre d'itérations d'EDP à appliquer. Ce schéma d'implémentation est particulièrement intéressant si on le compare aux schémas explicites usuels, basées sur les différences finies :

- Seules les dérivées premières ont besoin d'être estimées (dans le calcul du tenseur de structure lissé $\mathbf{G}_{\alpha,\sigma}$), alors qu'un schéma habituel de diffusion nécessite l'estimation numérique de dérivées secondes, plus sensibles aux données bruitées.
- Les structures d'images très fines sont mieux préservées d'un point de vue *numérique* : le lissage est effectué le long des lignes de champ de \mathbf{w} , avec une précision *sous-pixellique*, due à l'utilisation de schémas d'intégration numérique de Runge Kutta [PRESS92].

- Le pas de temps dt peut se permettre d'être important, notamment si la géométrie de lissage est considérée comme suffisamment régulière (c-à-d que son évolution au cours du temps sera négligeable). De ce point de vue, le schéma proposé est toujours stable, puisque le paramètre dt rentre en compte uniquement comme l'écart type de la pondération gaussienne dans la convolution le long des lignes de champ $\mathcal{C} \in \mathcal{F}$.
- Une conséquence immédiate est le fait que cet algorithme de lissage peut s'exécuter très rapidement, car très peu d'itérations (parfois une seule) sont nécessaires en pratique pour obtenir un résultat suffisamment régularisé, même si chaque itération est en elle-même plus complexe à calculer qu'avec un schéma explicite plus classique. Cela approxime bien sûr plus grossièrement la solution exacte de l'EDP Eq.(3.9), car on néglige la variation temporelle de la géométrie de lissage, par rapport à sa variation spatiale. Mais en réalité, cette supposition est souvent valable, notamment lorsque l'on traite des données à faible niveau de bruit.

Le choix du pas de discrétisation angulaire d_α est un paramètre relativement sensible de la méthode, puisque le lissage est en pratique calculé comme un moyennage de multiple *LIC* orienté par tous les angles α parcourant le plan 2D. Dans les régions où le lissage doit être fait de manière majoritairement anisotrope, seules quelques valeurs de α peuvent suffire, puisque dans ce cas, le lissage sera effectué le long d'une seule et unique direction, quelque soit la valeur de α . Par contre, dans les régions homogènes qui requièrent un lissage isotrope, les résultats seront d'autant plus précis que d_α sera faible, surtout si le pas de lissage inter-itération dt est important. En pratique, $d_\alpha = 30^\circ$, amène à une précision suffisante pour le lissage isotrope.

La Fig.3.2 illustre l'efficacité *numérique* de ce schéma, en le comparant au résultat que l'on pourrait obtenir avec une implémentation explicite plus classique basée sur le calcul des dérivées premières et secondes de l'image par différences finies. Les deux résultats sont obtenus par l'application de la même EDP Eq.(3.9) de diffusion anisotrope, avec les paramètres $p_1 = 0.01$ et $p_2 = 100$ (lissage très anisotrope, uniquement le long des contours couleurs θ_- , avec une amplitude de lissage partout égal à 1). Le schéma basé *LIC* (Fig.3.2c) préserve nettement mieux les structures fines, après un grand nombre d'itérations. L'avantage de la précision sous-pixellique du calcul des *LIC* sous-jacents prend ici tout son sens.

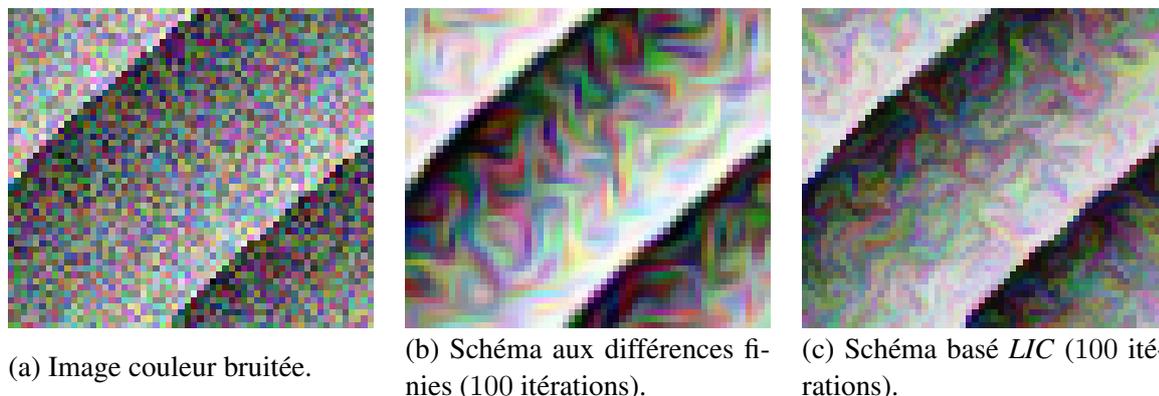


FIG. 3.2: Comparaisons entre schémas explicites aux différences finies et basés *LIC* pour l'implémentation de l'EDP Eq.(3.9).

3.3 Applications et résultats

Nous présentons ici quelques résultats d'application de notre EDP à préservation de courbure Eq.(3.9), implémentée à l'aide d'un schéma basé *LIC* et parallélisé (implémentation en *C++* et utilisation de la bibliothèque *OpenMP*). La méthode a été appliquée sur plusieurs images couleurs $\mathbf{I} : \Omega \rightarrow [0, 255]^3$, directement dans la base de couleur (R, G, B) . Toutes les expériences ont été réalisées sur un PC 2.8 Ghz à 24 cœurs. Pour chaque résultat présenté ci dessous, nous détaillons les paramètres utilisés ainsi que le temps de calcul.

Débruitage et lissage d'images couleurs :

- Le débruitage d'images est l'application la plus directe de notre méthode de régularisation.
- Fig.3.3 montre la suppression d'un motif dû à la numérisation d'un support comportant du grain (carte postale) sur une image de taille globale 586×367 (seule une petite portion est affichée). Notons que notre méthode est capable par lissage, de retirer ce bruit granuleux de taille importante tout en préservant les structures d'images significatives les plus fines (feuilles des palmiers par exemple). Là encore, une seule itération de notre EDP Eq.(3.9) a été nécessaire avec $p_1 = 0.5$, $p_2 = 0.7$, $\sigma = 1$ et $dt = 10$. Le temps de calcul est de 300 ms pour l'image entière.
 - La Fig.3.4 (première ligne) illustre le débruitage de l'image couleur "mandrill", artificiellement dégradée par un bruit blanc gaussien sur (R, G, B) (écart type $\sigma = 20$). Cette image couleur 512×512 a été régularisée avec l'Eq.(3.9) et une portion seulement de cette image est affichée. Une seule itération de notre EDP a été nécessaire, avec $p_1 = 0.5$, $p_2 = 0.7$, $\sigma = 1.5$ et $dt = 50$. Le temps de calcul est de 200 ms pour l'image entière.
 - La Fig.3.4 expose d'autres résultats de régularisation d'images, pour la suppression d'artéfacts de blocs après une compression JPEG, le débruitage d'images prises en mauvaise condition d'illumination, ou encore la création d'un effet peinture en exagérant le lissage.

Inpainting d'images par diffusion :

L'inpainting est une application de reconstruction, consistant à remplir de manière automatique des régions de l'image (définies par l'utilisateur) tel que la reconstruction semble la plus naturelle possible. C'est un problème difficile, que nous allons traiter plus en profondeur dans le chapitre 5, mais qui peut être résolu partiellement par l'utilisation d'EDP de diffusion anisotrope, telle que l'Eq.(3.9) présentée dans ce chapitre. L'idée est d'appliquer l'équation de diffusion seulement pour les pixels des régions à reconstruire, ce qui permet aux pixels voisins (connus) de se diffuser à l'intérieur de ces régions : une complétion non-linéaire de l'image le long des directions des isophotes entrantes dans le masque est ainsi naturellement réalisée. La Fig.3.5 illustre ce type d'application pour reconstruire des valeurs de pixels correspondant à la cage d'une perruche. Notons la bonne reconstruction des structures courbes de l'image, comme le bec de la perruche ou le contour de l'animal.

Agrandissement d'images :

Cette technique d'inpainting suggère naturellement qu'une EDP de diffusion peut être utilisée pour réaliser des interpolations non-linéaires d'images, et peut donc servir à agrandir une image (super-résolution), en remplacement des méthodes classiques d'interpolation linéaires (bi-cubiques ou Lanczos). Cela est réalisé de la façon suivante : on initialise la version agrandie de l'image avec une interpolation bicubique, puis on applique l'EDP à préservation de courbure Eq.(3.9) partout *sauf sur les points "connus" de l'image* (pixels dont la valeur a été utilisée pour le calcul de l'interpolation). C'est similaire à un processus d'inpainting dont le masque serait

défini comme le complémentaire à une grille parcimonieuse régulière. La Fig.3.6 illustre cette technique de super-résolution par EDP de diffusion, et compare les résultats obtenus avec les techniques d'interpolation linéaires d'images classiques (linéaire, bicubique). On aperçoit très nettement l'apport de la diffusion non-linéaire pour reconstruire des contours en super-résolution qui ne possèdent pas d'effets de crénelage.

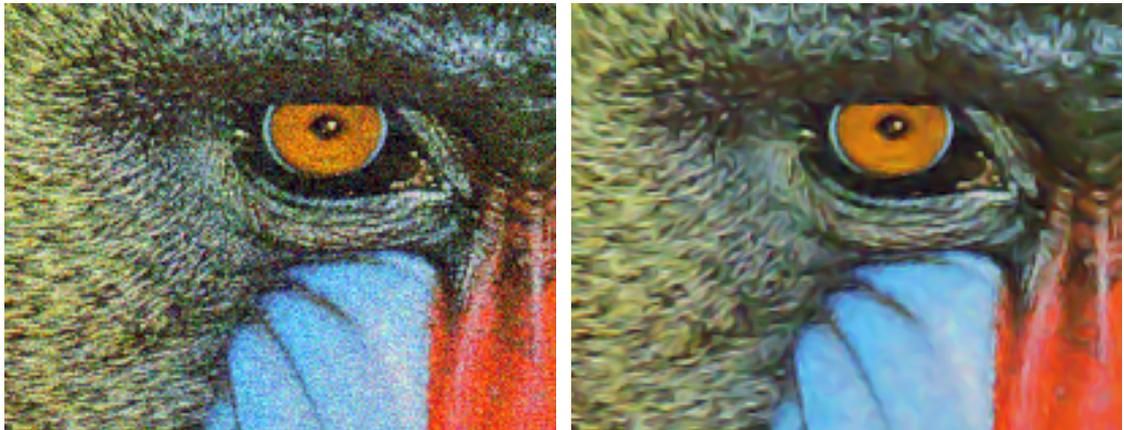
3.4 Conclusion du chapitre

La régularisation d'images multi-valuées est une étape qui reste fondamentale en traitement d'images, et il est primordial de pouvoir maîtriser le plus finement possible ce type de processus, pour le rendre flexible et adaptatif aux données et aux applications considérées. Dans ce chapitre, nous nous sommes attelés à cette tâche en proposant une EDP de diffusion (3.9) ayant la bonne propriété de préserver les structures courbes lors du processus de lissage. Cette EDP n'est pas seulement capable de préserver les détails fins et les structures courbées, elle peut surtout s'implémenter de manière rapide, élégante et efficace, puisque les schémas numériques associés sont capables de travailler à une précision sous-pixellique, en un temps de calcul réduit (ce dernier point étant l'un des inconvénients majeurs des méthodes itératives de régularisation d'images). Nous avons illustré toutes les propriétés de cette équation en l'appliquant sur un panel varié de traitements nécessitant du lissage anisotrope d'images.

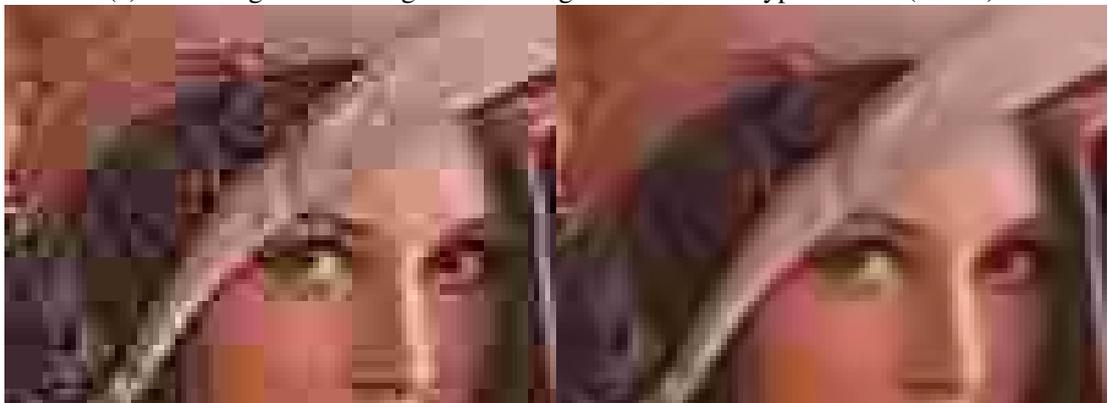
Le chapitre suivant continue dans cette lancée, en utilisant des techniques très similaires à celles développées ici, pour une application originale de rendu stylisé d'images sous forme de crayonnés.



FIG. 3.3: Suppression de bruit granuleux dans l'image "Tunisie" (300 ms).



(a) Débruitage d'une image avec bruit gaussien d'écart type $\sigma = 20$ (80 ms).



(b) Réduction d'artéfacts de compression JPEG sur l'image "Lena" compressée (100 ms).



(c) Création d'un effet "peinture" en exagérant le lissage (500 ms).

FIG. 3.4: D'autres résultats de lissage d'images couleurs par notre EDP à préservation de courbure Eq.(3.9).



FIG. 3.5: Application de l'EDP à préservation de courbure Eq.(3.9) pour l'inpainting d'images. Les pixels de la cage ont été manuellement labélisés comme faisant partie de l'ensemble des pixels à reconstruire.



Image couleur à agrandir

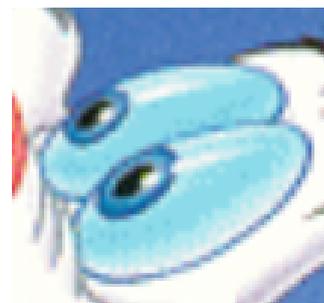


FIG. 3.6: Comparaisons de méthodes d'agrandissement d'images, utilisant des techniques d'interpolation linéaires (première ligne), bi-cubique (deuxième ligne) et basée EDP à préservation de courbure Eq.(3.9) (troisième ligne).

Un modèle tensoriel pour le rendu d'images en crayonnés

Sommaire

4.1 Transformations basées <i>traits</i>	40
4.2 Un algorithme de rendu crayonné d'images	42
4.3 Colorisation et extension artistique de l'algorithme	47
4.4 Conclusion du chapitre	53



Nous explorons ici une méthode de transformation automatique de photographies numériques sous forme de dessins ou de croquis stylisés. Nous élaborons un algorithme de rendu simple et rapide à mettre en œuvre, basé sur une simulation de tracés de traits de crayon dirigés par un champ de tenseurs d'ordre 2. La simulation cherche à reproduire (grossièrement) un processus de crayonnage qui aurait pu être exécuté par un artiste souhaitant reproduire une photographie couleur sous forme de croquis. Une étape de colorisation ultérieure, utilisant les couleurs de l'image d'entrée, permet de finaliser le rendu. Finalement nous peaufinons, avec l'aide d'un artiste, le schéma d'utilisation de notre algorithme pour améliorer de manière significative les rendus stylisés obtenus.

Période : 2011.

Publications associées :

- Conférence internationale : [C24] (ICIP'11).
 - Conférence nationale : [NC13] (GRETSI'11).
-

4.1 Transformations basées *traits*

Dans le domaine du rendu non-photoréaliste, les méthodes basées “traits” (*strokes* en anglais) [HERTZM03] servent à définir des processus de transformation d'images numériques, par exemple, sous forme de peintures [HERTZM98], croquis [DURAND01], gravures [DEUSSE00A], comics [DECARL02], ou encore de lavis [DEUSSE00B, SALISB96, WINKEN94]. Des techniques semblables ont été utilisées en visualisation d'images, pour la représentation de champs de vecteurs [CABRAL93], de tenseurs [TSCHUM03] ou d'images médicales [BRUCKN07]. Ces

algorithmes diffèrent principalement par le placement spatial ainsi que par la forme des primitives choisies comme briques de bases pour générer le rendu. Dans ce chapitre, nous proposons un algorithme de *simulation de traits de crayon*, ayant la particularité de générer à la fois une stylisation des zones de contours *et* des zones d'intensités plus homogènes. Ceci est rendu possible par l'utilisation de *champs de tenseurs du 2^{ème} ordre* qui permettent de modéliser finement la géométrie locale des traits que l'on souhaite tracer. Notre technique se base sur des "lancers" successifs de traits de crayon, à partir de positions aléatoires dans l'image, de telle manière que chacun de ces rendus soit consistant avec la géométrie tensorielle des traits prédéfinie, c-à-d en faisant apparaître des hachures spatialement régulières sur les régions homogènes (cas des tenseurs *isotropes*), et en surlignant nettement les contours lorsqu'il y en a (cas des tenseurs *anisotropes*). Notre méthode est particulièrement simple à implémenter et rapide à exécuter, tout en étant capable de générer de multiples styles différents de dessins, grâce à la variabilité naturelle du modèle tensoriel considéré (Fig.4.1). Des résultats additionnels d'application de notre algorithme après colorisation closent ce chapitre.

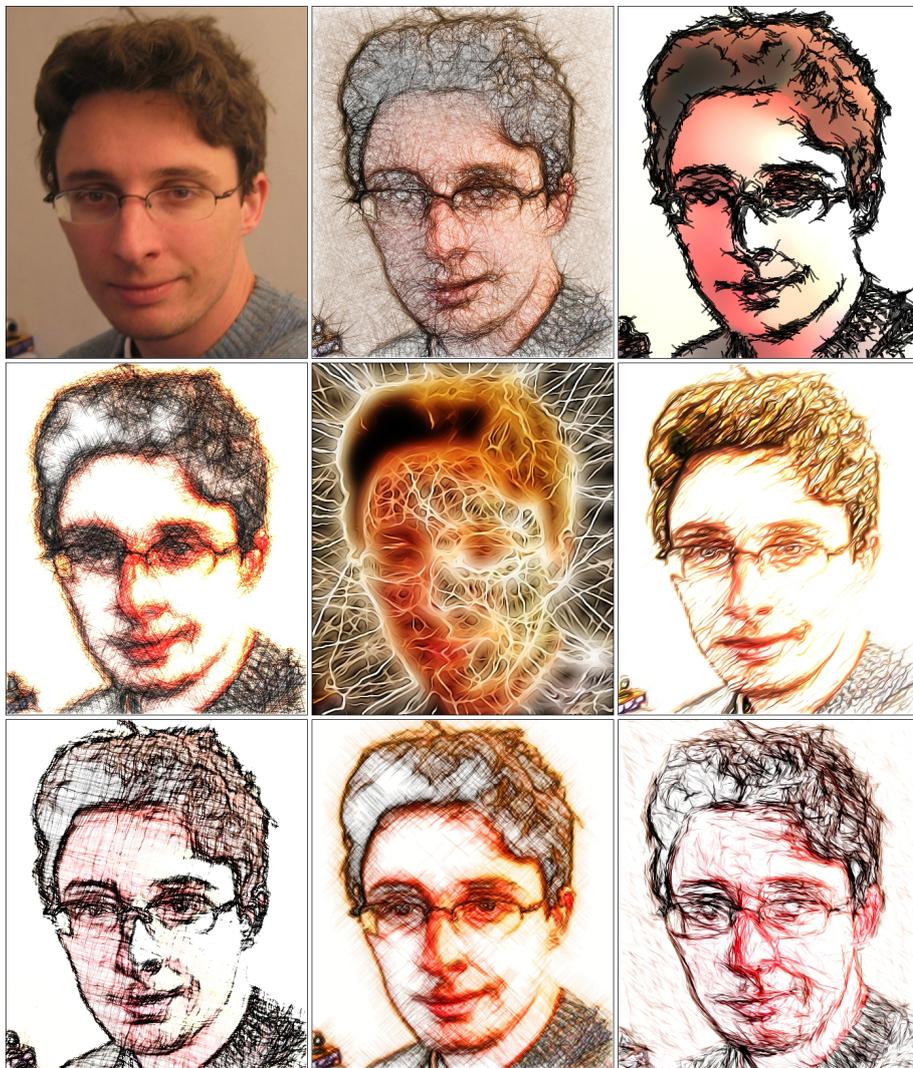


FIG. 4.1: Un algorithme de simulation de traits de crayon est proposé, pour le rendu stylisé automatique de photographies numériques couleurs. Notre méthode est capable de générer des styles de dessins variés à partir d'une même image d'entrée (ici, en haut à gauche).

4.2 Un algorithme de rendu crayonné d'images

4.2.1 Une première approche naïve pour le crayonné

Pour simuler la conversion d'une image couleur $\mathbf{I} : \Omega \rightarrow \mathbb{R}^3$ sous forme d'un crayonnage de traits noirs sur fond blanc, nous suggérons dans un premier temps de déterminer un ensemble de segments représentant les structures les plus significatives de \mathbf{I} . Comme la plupart des informations de contours sont présentes dans le gradient lissé de la composante de luminance $Y : \Omega \rightarrow \mathbb{R}$ de \mathbf{I} , on peut assez naturellement suggérer un premier algorithme naïf, pour la stylisation d'images par traits de crayon :

1. Calculer le champ de luminance Y de l'image couleur \mathbf{I} , puis le gradient de sa version lissée $\nabla Y_\sigma = Y * G_\sigma$, où G_σ est un noyau gaussien isotrope d'écart type $\sigma \in \mathbb{R}^+$.
2. Initialiser une image constante, à fond blanc $S : \Omega \rightarrow \mathbb{R}$, qui va contenir le rendu de l'algorithme de croquis automatique.
3. Tirer au sort une position $(x, y) \in \Omega$, telle que $\|\nabla Y_{(x,y)}\| \geq \epsilon$. Choisir un paramètre $\epsilon \in \mathbb{R}^+$ important permet ici de focaliser le rendu sur les contours les plus contrastés de l'image \mathbf{I} .
4. Tracer dans S le segment de couleur noire (partiellement transparent) de $(x - u, y - v)$ à $(x + u, y + v)$, où $(u, v) = \frac{L}{2} \frac{\nabla Y^\perp}{\|\nabla Y\|}$. La longueur du trait $L \geq 0$ a une valeur qui dépend de la résolution de l'image considérée, et peut être éventuellement pondérée par une fonction de $\|\nabla Y_{(x,y)}\|$ (contraste local).
5. Retour à l'étape 3, jusqu'à ce qu'un nombre significatif de traits aient été tracés dans S (dépend de la taille de Ω , et de L).



(a) Avec $(\epsilon, \sigma) = (0, 0)$

(b) Avec $(\epsilon, \sigma) = (20, 0)$

(c) Avec $(\epsilon, \sigma) = (20, 1)$

FIG. 4.2: Résultats de l'approche naïve, avec différents jeux de paramètres.

Cette approche rudimentaire illustrée en (Fig.4.2) permet déjà de générer des esquisses amusantes, mais où plusieurs caractéristiques “artistiques” typiques sont manquantes : D’abord, on ne permet que le tracer de traits de crayon *droits*, qui limitent la diversité de stylisation des contours (ainsi que l’intervalle de valeurs acceptables pour le choix de L). Ensuite, on ne génère pas de traits spatialement harmonieux dans les régions homogènes de l’image (c-à-d lorsque l’on choisit $\epsilon \rightarrow 0$). L’orientation $\frac{\nabla I}{\|\nabla I\|}$ n’ayant aucune cohérence dans ces zones ($\|\nabla I\| \rightarrow 0$), les segments résultants s’entrecroisent de manière quasi-aléatoire, contrairement à un motif de hachures par exemple, qui serait ici plus adapté. Dans la suite de ce chapitre, nous élaborons donc un *algorithme de simulation de traits dirigé par un modèle tensoriel*, permettant de maîtriser plus finement les comportements locaux de ces tracés de crayon.

4.2.2 Modélisation tensorielle de la géométrie des traits

Comme j’ai essayé de vous en convaincre dans les chapitres précédents, les tenseurs du second ordre adaptés pour modéliser des caractéristiques géométriques locales de natures isotropes ou anisotropes. Nous allons donc considérer ici de manière similaire, que les traits de crayon à simuler peuvent avoir des configurations soit isotropes (*hachures*), soit anisotropes (*contours*), selon que le point de départ du trait est situé sur une région quasi-homogène ou sur un contour de l’image. Modéliser cette variété de configurations par un champ tensoriel du second ordre semble donc assez adapté. L’approche en deux étapes que nous proposons est dans l’idée assez proche de celles définies dans [WEICKE99A, TSCHUM07] pour la régularisation d’images. Nous calculons, dans un premier temps, le champ de tenseurs de structure lissé :

$$\mathbf{G}_{\alpha,\sigma} = \left(\sum_{i=1}^n \nabla I_{i\alpha} \nabla I_{i\alpha}^T \right) * G_\sigma \quad \text{où } I_{i\alpha} = I_i * G_\alpha \quad (4.1)$$

En tout point $(x, y) \in \Omega$, les valeurs propres $\lambda_+, \lambda_- \geq 0$ de $\mathbf{G}_{\alpha,\sigma(x,y)}$ sont représentatives de la variation locale des couleurs dans \mathbf{I} . ($\lambda_+ + \lambda_-$) peut être alors considérée comme une mesure fiable pour déterminer le type de structure géométrique sur lequel (x, y) est situé (contour, région homogène ou coin). La base de vecteurs propres $\theta_+ \perp \theta_-$ donne quant à elle, l’orientation prédominante des structures locales (θ_- est parallèle au contour). La Fig.4.3a expose une illustration de $\mathbf{G}_{\alpha,\sigma}$ sous forme d’un champ d’ellipses de rayons $\lambda_{+/-}$ orientées par $\theta_{+/-}$. L’utilisation des tenseurs (4.1) assure que la corrélation inter-canaux est bien prise en compte (contrairement à l’approche naïve précédente), et ses paramètres de lissage permettent une estimation de la géométrie locale plus robuste au bruit et/ou à un espace échelle quelconque. Une fois que $\mathbf{G}_{\alpha,\sigma}$ est estimé, nous pouvons modéliser la géométrie des traits de crayon par *un autre champ tensoriel* $\mathbf{T} : \Omega \rightarrow \mathbb{P}(2)$ (*tenseurs de traits*) :

$$\forall (x, y) \in \Omega, \quad \mathbf{T}_{(x,y)} = c_+ \theta_+ \theta_+^T + c_- \theta_- \theta_-^T \quad (4.2)$$

avec $c_+ = \frac{1}{(1+\lambda_++\lambda_-)^{p_1}}$, $c_- = \frac{1}{(1+\lambda_++\lambda_-)^{p_2}}$ et $p_1 \geq p_2 \geq 0$.

Ces tenseurs de trait \mathbf{T} favorisent les comportements suivants :

- Lorsque $\mathbf{T}_{(x,y)}$ est anisotrope, il est orienté le long de $\theta_{-(x,y)}$, et le point (x, y) est probablement localisé sur un contour de l’image ($\lambda_{+/-}$ forts). Chaque trait de crayon partant de (x, y) devrait donc effectivement suivre la direction $\theta_{-(x,y)}$ du contour.

- Lorsque $\mathbf{T}_{(x,y)}$ est isotrope, le point (x, y) se trouve probablement sur une région de l'image quasi-homogène ($\lambda_{+/-}$ faibles), et les traits dessinés ici devraient générer des hachures, spatialement cohérentes, selon plusieurs directions du plan.
- Lorsque $\mathbf{T}_{(x,y)}$ définit une configuration intermédiaire (tenseurs “gonflés”, mais pas complètement isotropes), on souhaite qu'ils génèrent des traits représentant un mélange continu entre contours nets et hachures.

Les valeurs propres $c_{+/-}$ de $\mathbf{T}_{(x,y)}$ seront logiquement relatives à la longueur du trait tracé en (x, y) , à la fois pour les contours et les hachures, et dépendent plus ou moins du contraste local de \mathbf{I} . Notons qu'avec seulement deux paramètres p_1, p_2 pour \mathbf{T} , nous autorisons déjà une grande variété de styles de crayonnage, car on peut moduler de manière globale, le profil préférentiel des tenseurs (isotropes partout avec $p_1 \approx p_2$, anisotropes partout avec $p_1 \gg p_2 \gg 0$), ainsi que la dépendance des tenseurs au contraste local ($\lambda_+ + \lambda_-$). La Fig.4.3b expose un champ \mathbf{T} de tenseurs de trait, pour une image couleur, avec les paramètres $(p_1, p_2) = (1.2, 0.5)$ et $(\alpha, \sigma) = (0.5, 1.2)$.

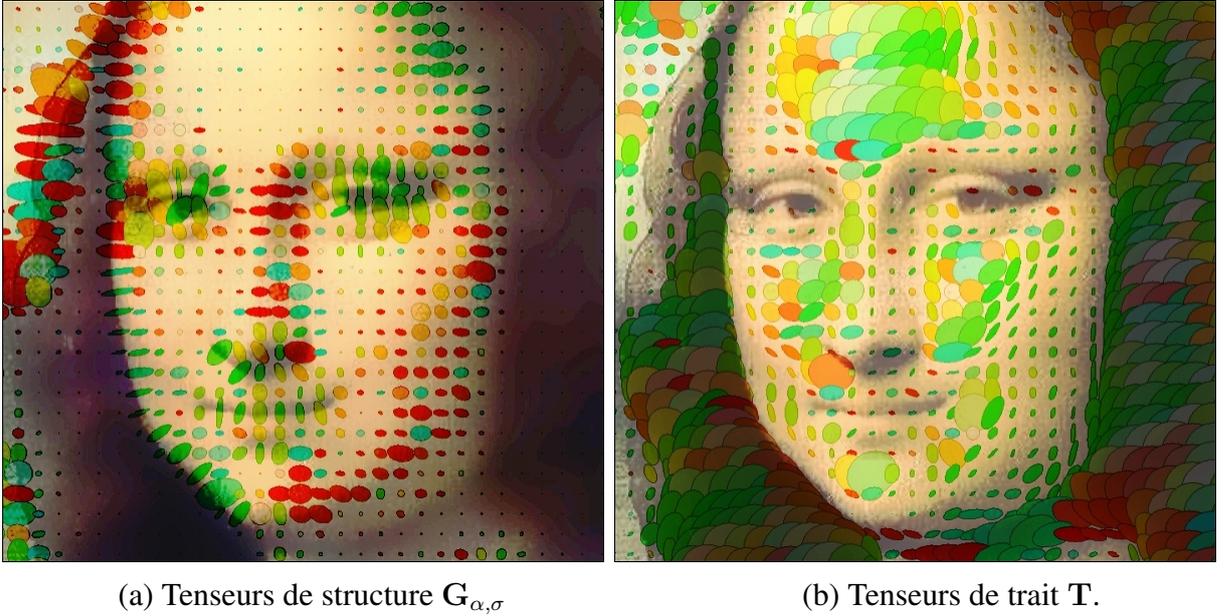


FIG. 4.3: Tenseurs de structure et tenseurs de traits, en surimpression de l'image couleur correspondante.

4.2.3 Lancer de traits guidés par des tenseurs

L'algorithme naïf de tracé de traits (section 4.2.1) ne sait gérer qu'une unique orientation $\nabla Y_{(x,y)} / \|\nabla Y_{(x,y)}\|$ pour tracer un segment en un point (x, y) . Ici, nous étendons cette technique à une géométrie tensorielle, ç-à-d le rendu d'un ou de plusieurs segments (éventuellement orientés différemment) selon que $\mathbf{T}_{(x,y)}$ est plutôt anisotrope ou isotrope. Pour cela, nous proposons de décomposer le champ de tenseurs de trait \mathbf{T} en plusieurs champs de vecteurs $\mathbf{w}_\gamma : \Omega \rightarrow \mathbb{R}^2$, exprimant les quotités de \mathbf{T} suivant les orientations $\gamma \in [0, \pi]$ du plan. En remarquant que

$$\int_{\gamma=0}^{\pi} a_\gamma a_\gamma^T d\gamma = \frac{\pi}{2} \mathbb{I}_d$$

avec $a_\gamma = (\cos \gamma \ \sin \gamma)^T$, nous pouvons écrire $\mathbf{T} = \frac{2}{\pi} \sqrt{\mathbf{T}} \left(\int_{\gamma=0}^{\pi} a_\gamma a_\gamma^T d\gamma \right) \sqrt{\mathbf{T}}$, et donc :

$$\mathbf{T} = \frac{2}{\pi} \int_{\gamma=0}^{\pi} \mathbf{w}_\gamma \mathbf{w}_\gamma^T d\gamma$$

avec $\mathbf{w}_\gamma = \sqrt{\mathbf{T}} a_\gamma$ ($\sqrt{\mathbf{T}}$ dénote ici la racine carré matricielle de \mathbf{T}).

Chaque $\mathbf{w}_\gamma \mathbf{w}_\gamma^T$ est un champ tensoriel unitaire purement anisotrope, orienté selon \mathbf{w}_γ .

- Lorsque $\mathbf{T}_{(x,y)}$ est parfaitement anisotrope ($c_+ \approx 0$), alors $\mathbf{w}_{\gamma(x,y)} = (\theta_- \cdot a_\gamma) c_-^{\frac{1}{2}} \theta_-$ est toujours parallèle à θ_- . Un trait de crayon orienté par \mathbf{w}_γ sera par conséquent toujours tracé dans la direction du contour, quelque soit l'angle γ considéré.
- Lorsque $\mathbf{T}_{(x,y)}$ est parfaitement isotrope ($c_+ \approx c_-$), alors $\mathbf{w}_{\gamma(x,y)} = c_- a_\gamma$ reste orienté par a_γ , ce qui veut dire que dans les régions quasi-homogènes, de multiples traits orientés différemment suivant les \mathbf{w}_γ vont être tracés à partir d'un même point (x, y) . On permet donc ici de créer un rendu localement *hachuré*, et spatialement cohérent, avec un type de hachures paramétrable, et uniquement dépendant de la façon dont γ est discrétisé (c-à-d par trois paramètres $\gamma_{min}, \gamma_{max}$ et $\delta\gamma$).

- Lorsque $\mathbf{T}_{(x,y)}$ définit une configuration intermédiaire, il va plus ou moins *attirer* les orientations a_γ vers ses vecteurs propres $\theta_{+/-}$, suivant son degré d'anisotropie. Cela va permettre la génération de transitions continues entre les deux configurations extrêmes correspondants aux cas isotropes (hachures) et anisotropes (contours).

La décomposition de \mathbf{T} en de multiples champs vectoriels \mathbf{w}_γ suggère également d'exprimer un trait non plus comme un simple segment orienté par \mathbf{w}_γ , mais comme la ligne de champ $\mathcal{C} : [-\frac{L}{2}, \frac{L}{2}] \rightarrow \mathbb{R}^2$ de \mathbf{w}_γ , partant de (x, y) et de longueur L :

$$\frac{\partial \mathcal{C}(l)}{\partial l} = \mathbf{w}(\mathcal{C}(l)), \quad \text{où } \mathcal{C}(0) = (x, y) \quad (4.3)$$

Considérer ces lignes de champ comme primitives de trait va permettre de mieux "coller" aux contours des images, quelque soit leur courbure. Les rendus de stylisation qui en découlent paraissent ainsi plus délicats. Finalement, notre algorithme de simulation de tracé de traits à géométrie tensorielle s'écrit :

1. Calculer le champ de tenseurs de trait \mathbf{T} (4.2) à partir de la photographie numérique couleur \mathbf{I} .
2. Décomposer \mathbf{T} en plusieurs champs de vecteurs $\mathbf{w}_\gamma = \sqrt{\mathbf{T}} a_\gamma$, pour γ discrétisé dans $[\gamma_{min}, \gamma_{max}]$ avec un pas $\delta\gamma$.
3. Initialiser une image constante à fond blanc $S : \Omega \rightarrow \mathbb{R}$ qui contiendra le rendu final du crayonnage (en niveau de gris).
4. Tirer au sort une position (x, y) dans Ω , telle que $\sqrt{\lambda_+ + \lambda_-} \geq \epsilon$. Notons que ϵ peut maintenant être ici choisi proche de 0.

5. Pour chaque γ , tracer dans S le segment (partiellement opaque) de $(x, y) - L \mathbf{w}_{\gamma(x,y)}$ à $(x, y) + L \mathbf{w}_{\gamma(x,y)}$, ou la ligne de champ (4.3) de \mathbf{w}_{γ} , partant de (x, y) , de longueur $L\mathbf{w}_{\gamma(x,y)}$.
6. Retour à l'étape 4, jusqu'à ce qu'un nombre significatif de traits aient été tracés dans S .

Une illustration du fonctionnement de l'algorithme est présentée en Fig.4.4 sur une image synthétique comprenant des parties distinctes isotropes et anisotropes, avec des primitives en forme de lignes de champs. Notons la façon dont différents styles de hachures sont générés à partir de différents jeux de paramètres $(\gamma_{min}, \gamma_{max}, \delta_{\gamma})$.

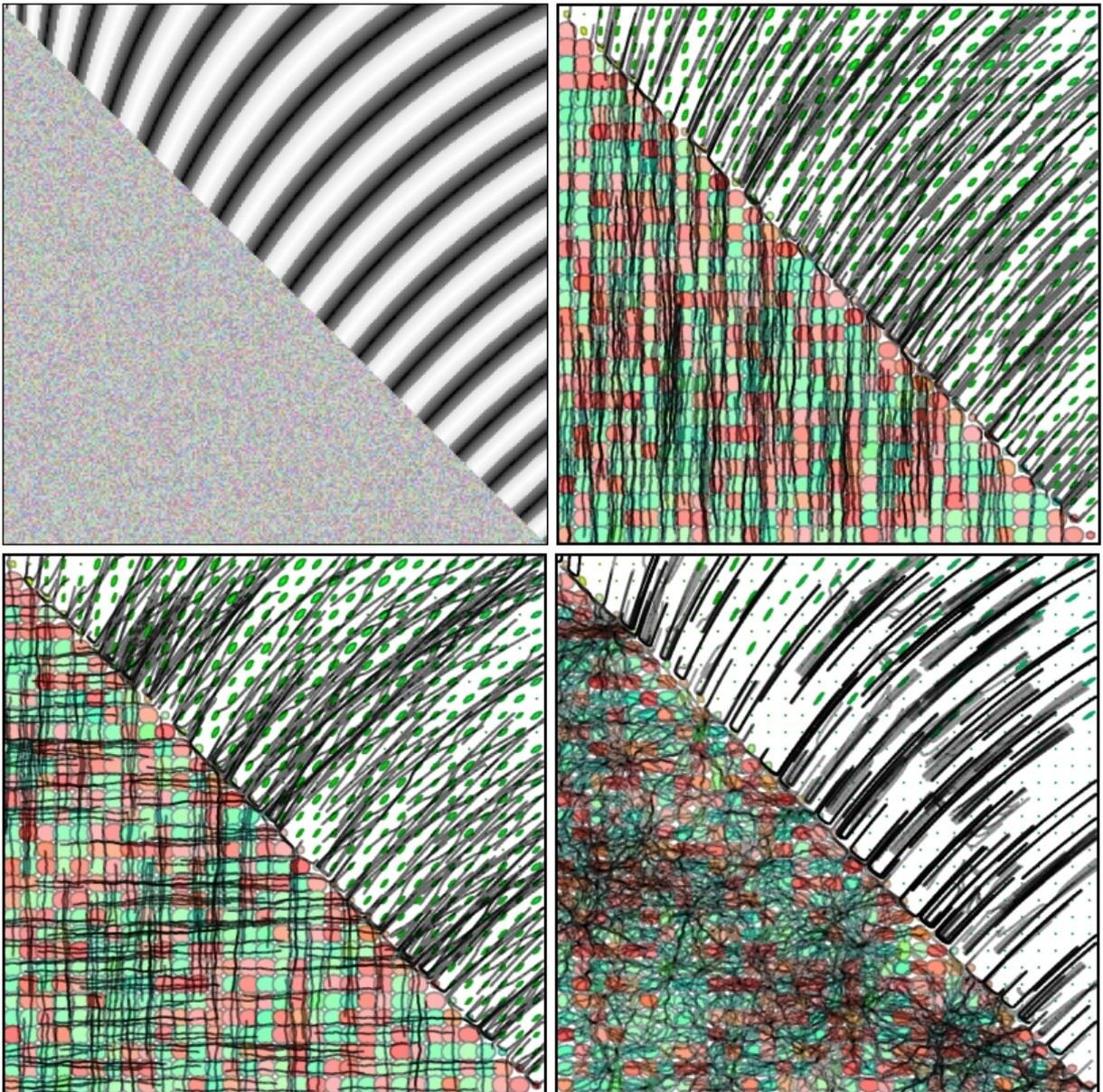


FIG. 4.4: Simulation dirigée tenseurs de traits de crayon, à partir d'une image synthétique (en haut à gauche) contenant à la fois des structures isotropes (bruit) et anisotropes (bandes orientées). Les paramètres $(\gamma_{min}, \gamma_{max}, \delta_{\gamma})$ sont $(90^\circ, 90^\circ, 0)$ (en haut à droite), $(0^\circ, 90^\circ, 90^\circ)$ (en bas à gauche) et $(15^\circ, 165^\circ, 20^\circ)$ (en bas à droite).

4.3 Colorisation et extension artistique de l'algorithme

Une fois le rendu du croquis en niveau de gris obtenu, nous pouvons optionnellement le composer avec les couleurs originales de l'image d'entrée pour générer un rendu de dessin final, coloré. Pour générer les résultats présentés dans les figures suivantes, nous avons appliqué des techniques de composition simples de calques, telles que celles utilisées en routine par les infographistes (*multiplication, overlay, hard light, . . .*). Ces traitements additionnels sont triviaux, et peuvent se réaliser de manière complètement automatique, l'étape essentielle de transformation de l'image étant réalisée par notre algorithme de simulation de trait de crayon. Cette technique de crayonnage automatique s'avère particulièrement flexible et générique, et sait générer des rendus de croquis intéressants, où les structures contenues dans des photographies couleurs sont naturellement extraites et mises en avant. L'algorithme est simple et peut être potentiellement parallélisé pour des applications temps réelles, avec des perspectives intéressantes pour l'application de conversion automatique de vidéos réelles en dessins animés par exemple. Les Fig.4.5 à 4.7 illustrent l'application directe de notre algorithme sur des images couleurs d'entrées, et la souplesse que permet l'algorithme proposé pour le rendu stylisé d'images.

Notons également que cet algorithme simple a pu être pris en main rapidement par certains utilisateurs de la plateforme logicielle *G'MIC*, plateforme libre que nous développons au laboratoire GREYC et qui est utilisée par beaucoup d'artistes amateurs et professionnels à travers le monde (plus de détails sur ce logiciel seront donnés en chapitre 8). Nous avons eu notamment l'occasion de collaborer avec *Tom Keil*, l'un de ces utilisateurs enthousiastes, qui a longtemps expérimenté l'algorithme de base et l'a pris en main pour adapter le rendu à ses besoins. La Fig.4.8 illustre la façon dont cet artiste a utilisé notre algorithme pour la stylisation d'images (*workflow*) : son idée consiste à appliquer notre méthode 4 fois consécutivement sur la même image de départ, en choisissant des paramètres différents pour obtenir à chaque fois un comportement différent de la simulation des traits de crayon, spécifiquement pour faire ressortir les zones suivantes :

- **Highlights** : Les zones d'illumination importantes de l'image.
- **Shadows** : A l'inverse, les zones qui correspondent aux régions sombres de l'image.
- **Outlines** : Un rendu spécifique est effectué pour isoler les contours saillants de l'image (tenseurs de trait nuls partout sauf sur les contours).
- **Details** : Un autre jeu de paramètre est utilisé pour intégrer contours et textures, et ignorer les zones homogènes.

Les 4 images obtenues sont alors masquées et combinées de manière simple, pour réaliser le rendu final de l'image en noir et blanc. Dans un dernier temps, ce rendu est colorisé par les couleurs de l'image d'origine. Il est important de noter que ces étapes additionnelles peuvent être entièrement automatisées, car elles ne nécessitent pas d'analyse géométrique supplémentaire de l'image d'entrée. La Fig.4.8 (dernière ligne) montre la différence de rendus obtenus par ce pipeline étendu et l'application de notre algorithme de base directement sur l'image originale.

La Fig.4.9 expose d'autres résultats de stylisation d'images couleurs obtenus par cet artiste avec l'application de sa proposition d'extension de notre algorithme de simulation de crayonnés.

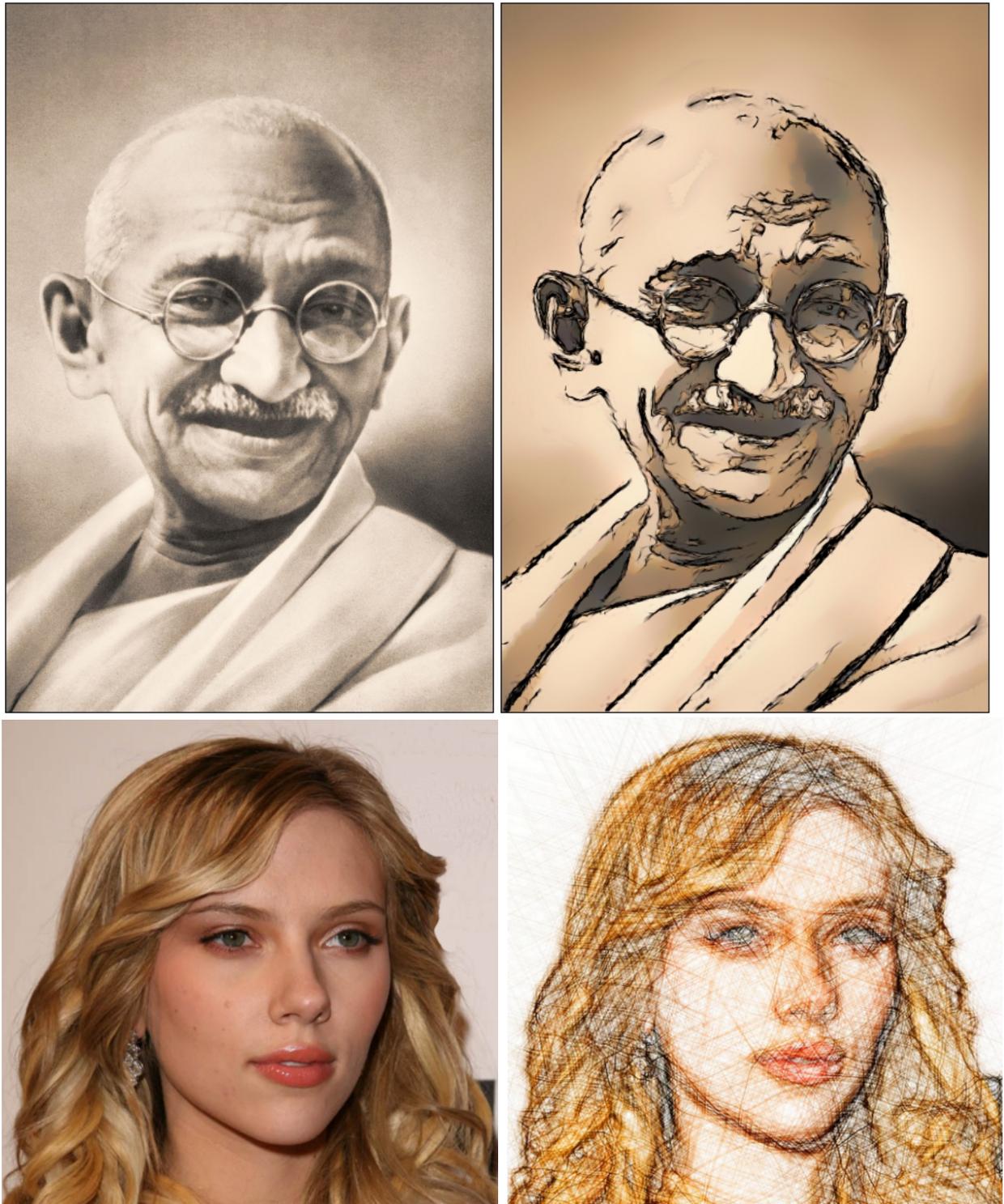


FIG. 4.5: Illustration de notre algorithme de stylisation d'images, avec des champs de tenseurs de traits différents : 1. très anisotrope sur les pixels de contours, et nul ailleurs (première ligne), ou 2. isotrope partout mais de valeurs propres faibles sur les zones homogènes.



FIG. 4.6: Autres exemples de stylisation d'images obtenus par l'algorithme proposé. Notons que pour l'image du crâne, seule la structure de l'image d'origine est utilisée ici (les couleurs ne le sont pas).

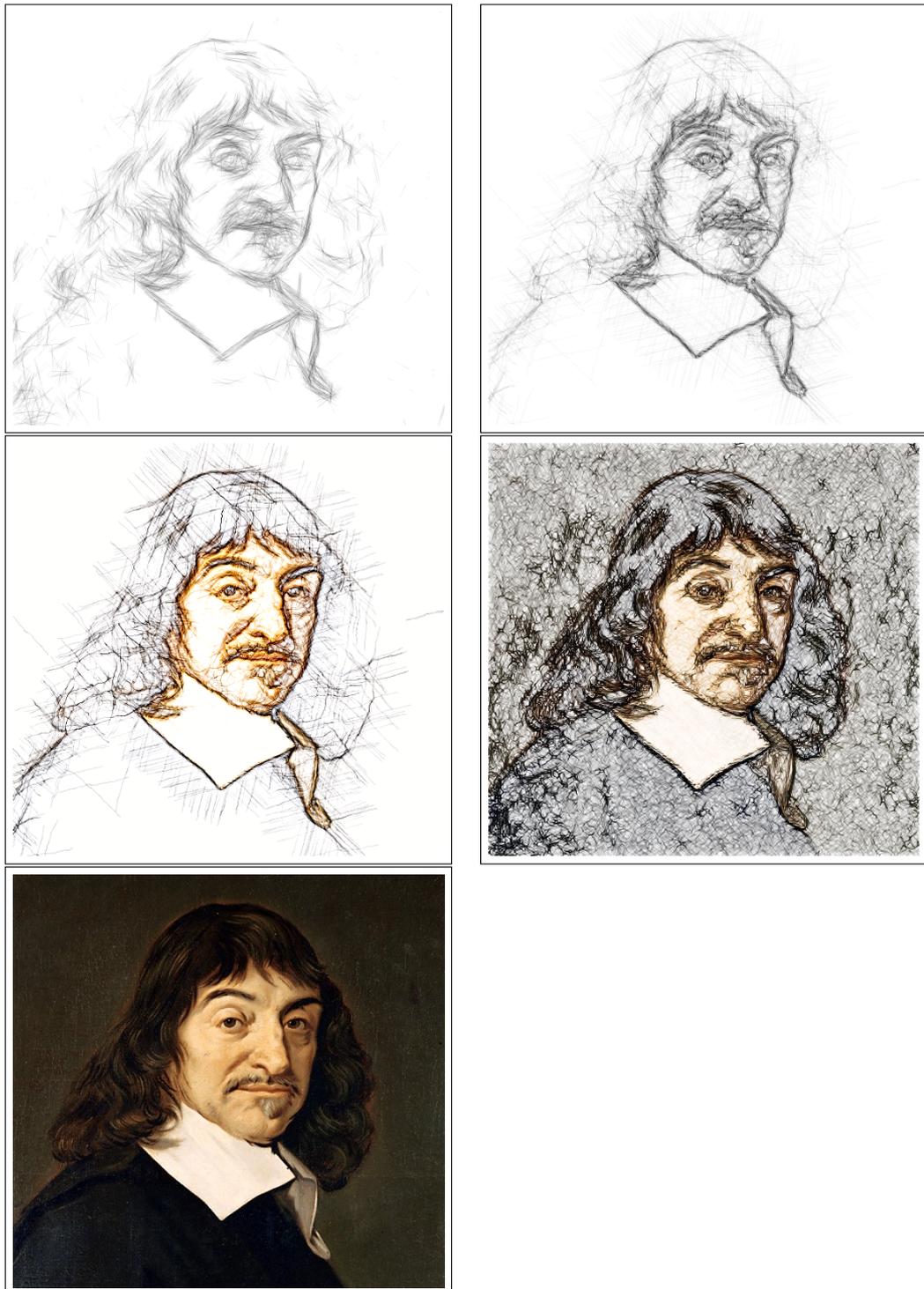
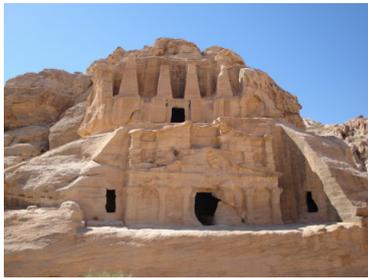
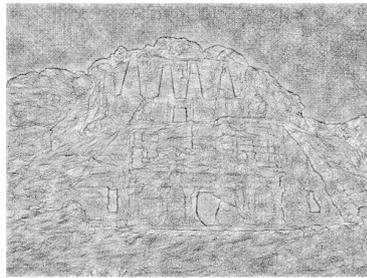
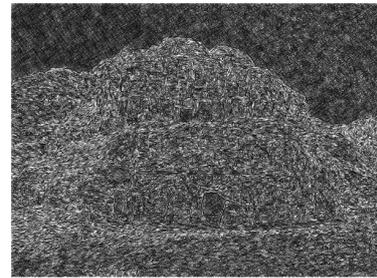
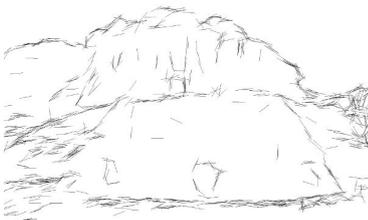
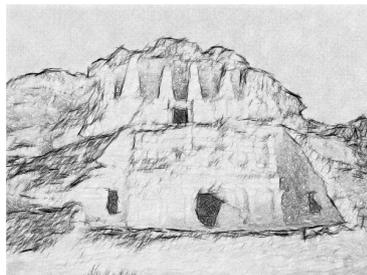


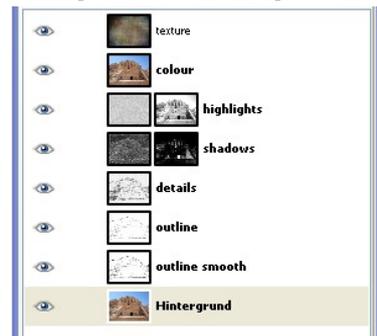
FIG. 4.7: Simulation d'étapes chronologiques de croquis pour arriver à une peinture finale (dernière image, qui est la seule image d'entrée utilisée par notre algorithme dans chacune des étapes simulées).



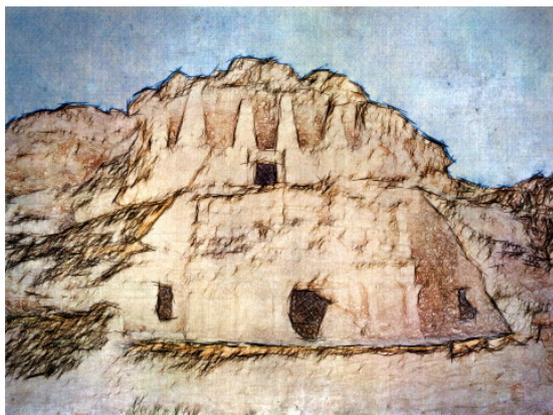
a) Photographie de référence.

b) Résultat pour les *highlights*, à partir de a).c) Résultat pour les *shadows*, à partir de a).d) Résultat pour les *outlines*, à partir de a).e) Masque de *highlight*, à partir de a).f) Masque de *shadow mask*, à partir de a).g) Résultat pour les *details*, à partir de a).

h) Résultat N&B des 4 croquis, après masquage.



i) Organisation finale des calques.



j) Résultat de la stylisation après fusion des images et colorisation



k) Résultat de l'algorithme direct sur l'image originale.

FIG. 4.8: Vue d'ensemble du *workflow* de l'artiste *Tom Keil* utilisant notre algorithme de rendu non-photoréaliste d'images sous forme de crayonnés.



FIG. 4.9: Autres exemples de stylisation d'images obtenus par application du *workflow* de l'artiste *Tom Keil*.

4.4 Conclusion du chapitre

Ce chapitre, parti d'une idée simple de modélisation tensorielle de la géométrie de traits de crayon, propose un algorithme rapide de rendu non-photoréaliste d'images transformant des images couleurs en croquis crayonnés. La collaboration avec un un artiste amateur *Tom Keil* nous a permis de définir un pipeline de stylisation d'images avancé, basé sur l'application successives de notre algorithme avec différents jeux de paramètres, et la fusion des images obtenues. Les résultats obtenus sont originaux et très encourageants pour une extension future à la transformation automatique de séquences vidéos en dessins animés. L'utilisation de tenseurs de second ordre pour la modélisation des structures géométriques anisotropes/isotropes montre une fois de plus sa pertinence pour la modélisation géométrique du processus à appliquer.

Tenseurs pour l’inpainting d’images basé motifs

Sommaire

5.1 L’inpainting d’images basé motifs	54
5.2 Quelques contributions à l’état de l’art	61
5.3 Résultats sur des images couleurs	73
5.4 Inpainting de paires stéréoscopiques avec carte de profondeur . .	74



Ce chapitre introduit la problématique de l’inpainting d’images (complétion/remplissage de zones d’images inconnues), et s’intéresse plus particulièrement aux méthodes basées motifs (patches). Après une revue rapide des méthodes existantes, nous proposons quelques améliorations significatives d’un algorithme de l’état de l’art, basées sur l’utilisation de champs tensoriels aussi bien pour l’analyse de la géométrie locale de l’image à compléter que pour sa synthèse par recopie de patches. Nous étendons par la suite ces améliorations pour la complétion de séquences vidéos et de paires stéréoscopiques resynthétisées (vues virtuelles) avec cartes de profondeur.

Période : 2013–2015.

Encadrements : Maxime Daisy (doctorant, 3 ans),

Pierre Buysens (post-doctorant, 2 ans).

Financements : Projet Action 3DS (Projet “Grand emprunt”, 9 partenaires).

Publications associées :

– Journal : [J10].

– Conférences internationales : [C32], [C31] (Siggraph Asia’15), [C30] (CAIP’15), [C29] (ICIP’15), [C28] (ICIP’14), [C27] (Siggraph Asia’13), [C26] (CAIP’13).

– Conférence nationales : [NC19], [NC18], [NC17] (GRETSI’15), [NC16] (CORE-SA’13), [NC14] (GRETSI’13).

5.1 L’inpainting d’images basé motifs

L’Inpainting aussi connu sous les noms de désocclusion, complétion ou remplissage intelligent d’images, se rapporte au problème de la reconstruction de parties manquantes (ou abîmées)

dans une image, de telle sorte que l’image reconstruite ait un aspect aussi réaliste ou naturel que possible.

La particularité de ce problème est de devoir reconstruire des zones de données entières complètement inconnues (souvent nommées *masque* ou *trou*). De nombreuses applications de l’*inpainting* ont émergé ces dernières années : la restauration d’images contenant des rayures, la suppression de logos ou de texte, la super-résolution, la suppression d’objets ou de personnes, en sont quelques exemples. Depuis le début des années 2000, une recherche active a été menée sur ce sujet, et a débouché sur de nombreux algorithmes différents, dont des revues assez complètes peuvent être trouvées dans [GUILLE14] et [BUYSSE15B].

Les méthodes existantes peuvent être divisées en deux types d’approches : les approches *géométriques* (purement locales), et les approches *basées patch*.

- Les approches *géométriques* sont les plus anciennes [MASNOU02, BERTAL00, TSCHUM05] et cherchent à compléter les images en utilisant des interpolations géométriques locales. Ces méthodes produisent des résultats intéressants (voir chapitre 3), mais ne reproduisent souvent aucune texture.
- Les approches *basées patch* utilisent, comme leur nom l’indique, une modélisation de l’image comme un ensemble de patches à la fois pour l’analyse du contenu et pour la synthèse du résultat. Ces dernières approches incluent à la fois des techniques *parcimonieuses* (uniquement pour l’analyse) et des techniques *basées motifs* (pour l’analyse et la synthèse, en proposant généralement des algorithmes de copier/coller de patches plus ou moins complexes). Ces techniques sont plus efficaces que les approches géométriques lorsque le masque à remplir est large.

Nous nous focalisons ici sur cette dernière classe de méthode, avec trois contributions principales :

- En premier lieu, nous proposons une revue des méthodes d’*inpainting* *basées motifs*. En particulier, l’algorithme pionnier de Criminisi *et al.* [CRIMIN04] est détaillé et une revue technique des travaux dérivés de cette technique est proposée.
- Puis, nous proposons quelques ajustements pour améliorer le rendu de cet algorithme, en particulier en formalisant une extension “dirigée tenseurs” d’une méthode de fondu spatial de patches, qui a l’avantage de réduire fortement les artéfacts de reconstructions typiques de ce type de méthodes.
- Enfin, nous étendons les algorithmes étudiés pour la reconstruction d’images 2D au cas de paires stéréoscopiques correspondant à des vues virtuelles resynthétisées, avec prise en compte des cartes de profondeur associées.

Plus de détails sur l’état de l’art peut être trouvé dans le manuscrit de thèse de *Maxime Daisy* [DAISY15].

Les techniques d’*inpainting* basées motifs utilisent les patches d’une image à la fois pour *analyser* leur contenu de la manière la plus intelligente possible, et pour *reconstruire* les parties manquantes. Ces méthodes peuvent être subdivisées elles-mêmes en trois approches principales :

- Les approches *gloutonnes* [BORNAR02, DRORI03, CRIMIN04, PÉRE04, LE MEU11, XU10, MARTIN12] qui remplissent le masque en une seule passe, en copiant/collant de multiples patches de géométrie variées de manière incrémentale.
- Les approches *hybrides* [BERTAL03, STARCK05, JIA04, CAO11] qui incorporent des éléments des méthodes *géométriques* afin de reconstruire dans un premier temps les macrostructures, puis les textures dans un deuxième temps.

- Les méthodes *basées énergie* [WEXLER07, KOMODA07, PRITCH09, MANSFII1, ARIAS12, HE12A, NEWSON13] qui cherchent le plus souvent à minimiser une énergie basée sur les données connues de l'image, et qui requièrent plusieurs itérations/passes pour converger vers un résultat satisfaisant. Ces méthodes reconstruisent généralement les parties manquantes au niveau pixel en mélangeant des patchs d'images ensemble.

Notations spécifiques. Une image à reconstruire sera considérée comme une fonction $I : \mathcal{I} \rightarrow \mathbb{R}^3$ (image couleur) où \mathcal{I} définit le domaine de l'image Ω est la partie masquée de l'image que l'on cherche à reconstruire (c-à-d l'ensemble des pixels inconnus), et $\delta\Omega$ dénote le bord du masque. Dans ce chapitre, un patch Ψ_p centré en un pixel p est considéré comme étant une fonction $\Psi_p : \mathcal{N}_p \rightarrow \mathbb{R}^3$ où $\mathcal{N}_p \subset \mathcal{I}$ est le support carré de Ψ_p . Notons que ce patch peut être lui-même masqué (c-à-d que certains de ses pixels peuvent être inconnus). $\Psi_{\hat{p}}$ dénote le patch qui est le plus proche de Ψ_p selon une métrique donnée d :

$$\Psi_{\hat{p}} = \left\{ \Psi_q \mid \arg \min_{q \mid \text{Card}(\mathcal{N}_q \cap (\mathcal{I} - \Omega)) = \text{Card}(\mathcal{N}_p)} d(\Psi_p, \Psi_q) \right\}$$

La distance d la plus simple (et la plus utilisée, principalement pour des raisons de rapidité de calcul) pour comparer la similarité visuelle entre deux patchs est la *somme des différences au carré* (SSD) :

$$d_{SSD}(\Psi_p, \Psi_q) = \sum_{v \in (\mathcal{N}_p \cap (\mathcal{I} - \Omega))} \|\Psi_p(v) - \Psi_q(v + p - q)\|^2 \quad (5.1)$$

Plusieurs autres distances ont été proposées dans la littérature, telle que la distance de Bhattacharya [LE MEU13A] ou la distance de Hellinger [MARTIN12], associées à la SSD pour comparer des fonctions de densités de probabilité de patchs. Néanmoins, il n'est pas évident de montrer que ces distances aient un intérêt particulier par rapport à l'utilisation de la SSD seule, notamment dans le cas où l'inpainting doit être appliqué à des images quelconques.

Approches gloutonnes. Les approches *gloutonnes* consistent en le remplissage graduel de Ω avec des pixels ou des groupes de pixels jusqu'à ce que $|\Omega| = \emptyset$, chaque valeur de pixel de Ω étant donc assignée une seule fois. Ces méthodes suivent toujours le même schéma, pouvant se décomposer en 4 étapes principales :

1. Sélection d'un pixel p appartenant au bord du masque $\delta\Omega$,
2. Recherche du patch $\Psi_{\hat{p}}$ le plus similaire au patch Ψ_p centré en p ,
3. Copie des valeurs connues du patch $\Psi_{\hat{p}}$ dans le voisinage inconnu de p (dans Ω).
4. Si $|\Omega| \neq \emptyset$, revenir à l'étape 1.

Les deux premières étapes sont d'une importance critique pour les résultats d'inpainting. Les implémentations naïves aboutissant souvent à des résultats non-satisfaisants, nous détaillons plus précisément les points essentiels de ces deux étapes pour l'algorithme de Criminisi *et al.* [CRIMIN04].

Ordre de remplissage. Chacun des trois papiers originaux décrivant ce type d'algorithme d'inpainting [BORNAR02], [DRORI03], et [?] met l'accent sur l'importance de l'ordre de remplissage (c-à-d de la sélection du pixel p sur le bord du masque $\delta\Omega$). Le remplissage couche par couche, en partant du bord du masque d'inpainting jusqu'à son centre (appelé remplissage en *pe-lure d'oignon*), est utilisé dans [BORNAR02] et selon un schéma multi-échelle dans [DRORI03]. Cet ordre de remplissage peut sembler naturel, mais génère souvent des résultats à l'aspect non

naturel, particulièrement près du centre du masque : les structures importantes qui auraient dû être reconstruites d'abord sont généralement perdues.

Pour pallier ces difficultés, Criminisi *et al.* ont proposé dans [CRIMIN04] un ordre de remplissage basé sur l'analyse des structures traversant $\delta\Omega$: un terme de priorité P_p est calculé pour chaque pixel de $p \in \delta\Omega$, de la façon suivante :

$$P_p = C_p \cdot D_p \quad (5.2)$$

où C_p est une mesure dite de *confiance*, et D_p est définie comme un *terme de données* qui prend en compte la présence des structures dans Ψ_p . Ce dernier terme joue un rôle très important dans le calcul de la priorité de remplissage, et favorise la continuation de structures (contours) qui entrent dans le masque Ω . Plus précisément, le terme de *confiance* C_p peut être vu comme une mesure du nombre de données fiables qui se trouvent dans le voisinage de p . Il est défini dans [CRIMIN04] comme étant :

$$C_p = \frac{\sum_{q \in (\mathcal{N}_p \cap (\mathcal{I} - \Omega))} C_q}{|\mathcal{N}_p|} \quad (5.3)$$

où $|\cdot|$ est la taille de \mathcal{N}_p (c-à-d., son nombre de pixels), et lors de l'initialisation de l'algorithme on suppose que :

$$\begin{cases} C_p = 1 & \forall p \in \mathcal{I} - \Omega \\ C_p = 0 & \forall p \in \Omega \end{cases}$$

Ce terme a des valeurs importantes près des bords du masque initial, et décroît petit à petit près du centre de Ω . Il tend à faire reconstruire en priorité les pixels qui ont le plus d'information valide dans leur voisinage. C'est un terme similaire qui définit l'ordre de remplissage en *pelure d'oignon* dans [BORNAR02, DRORI03].

Le *terme de données* D_p reflète quant-à-lui l'importance des structures locales d'images aux alentours de Ω . Il est défini dans [CRIMIN04] par :

$$D_p = \frac{|\overrightarrow{\nabla I_p^\perp} \cdot \vec{n}_p|}{\alpha} \quad (5.4)$$

avec $\overrightarrow{\nabla I_p^\perp}$ la direction locale de l'isophote principale, définie comme :

$$\overrightarrow{\nabla I_p^\perp} = \left\{ \overrightarrow{\nabla I_q^\perp} \mid \arg \max_{q \in (\mathcal{N}_p \cap (\mathcal{I} - \Omega))} \|\overrightarrow{\nabla I_q}\| \right\}, \quad (5.5)$$

\vec{n}_p est le vecteur normal à Ω en p , α est une constante de normalisation (qui peut être en réalité ignorée puisque c'est la même pour tous les p), et $\overrightarrow{\nabla I_p}$ est le vecteur gradient couleur en p (vecteur gradient de norme maximale). Si plusieurs vecteurs gradients ont la même valeur maximale de norme, on choisit celui qui maximise $|\overrightarrow{\nabla I_p^\perp} \cdot \vec{n}_p|$. Ce terme de données favorise la reconstruction de structures locales linéaires qui traversent le masque Ω de manière orthogonale, et on retrouve donc la notion d'isophote chère aux méthodes *basées géométrie*. Ce terme affecte fortement l'ordre de remplissage, en permettant de reconstruire des structures qui auraient été perdues avec un schéma plus classique de remplissage *en pelure d'oignon*.

Une fois que toutes les priorités $P_q, \forall q \in \delta\Omega$ ont été calculées, le pixel possédant la priorité la plus haute est choisi comme le pixel *destination* p à reconstruire.

Recherche du meilleur patch et reconstruction. A chaque itération, on recherche dans l'image le patch $\Psi_{\hat{p}}$ le plus assorti avec la partie non-masquée du patch Ψ_p centré au pixel destination p . Au lieu de chercher dans toute l'image, on réduit en général la zone de recherche à un carré de taille plus petite, centré en p afin de réduire le temps de calcul. Plus d'informations sur cette étape clé sont données à la fin de cette section. Une fois le patch $\Psi_{\hat{p}}$ trouvé, sa partie non masquée est redessinée en p dans $\mathcal{N}_p \cap \Omega$. Les valeurs du terme de confiance sont également dupliquées, à partir de la confiance en Ψ_p :

$$C_q = C_p, \forall q \in \mathcal{N}_p \cap \Omega \quad (5.6)$$

Améliorations proposées dans l'état de l'art. De nombreuses variations de l'algorithme original ont été proposées dans la littérature. Puisque le processus d'inpainting est glouton, un changement mineur dans la priorité de remplissage a potentiellement des répercussions significatives sur le résultat final reconstruit. Il n'est donc pas étonnant que plusieurs travaux aient été conduits dans le but de calculer des termes de priorités les plus efficaces possibles.

- Les auteurs de [LE MEU11] ont proposé par exemple l'utilisation d'heuristiques particulières pour améliorer la priorité de remplissage en utilisant un terme basé tenseur :

$$D_p = \alpha + (1 - \alpha) \exp\left(\frac{\eta}{(\lambda_1 - \lambda_2)^2}\right) \quad (5.7)$$

où λ_1 et λ_2 sont les valeurs propres du tenseur de structure [DI ZEN86], η et α sont des hyper-paramètres fixés arbitrairement par les auteurs ($\eta = 8$ et $\alpha = 0.01$). L'utilisation des tenseurs apporte une meilleure modélisation des variations locales de l'image.

- Les auteurs de [XU10] ont proposé un terme de données parcimonieux qui mesure pour chaque $p \in \delta\Omega$ la similarité globale d'un patch centré en p avec les patches contenus dans un voisinage pré-défini. L'éparpillement d'un pixel de bord est calculé comme :

$$\rho_p = \|\vec{\omega}_p\|_2 \sqrt{\frac{|N_s(p)|}{|N(p)|}} \quad (5.8)$$

où $|N_s(p)|$ est le nombre de patches valides dans le voisinage de p (c-à-d les patches qui ne contiennent aucun pixel masqué), $|N(p)|$ est le nombre total de patches dans le voisinage de p , et $\vec{\omega}_p$ est le vecteur qui contient les similarités w_{p,p_j} , défini comme :

$$w_{p,p_j} = \frac{1}{Z_p} \exp\left(\frac{-d(\Psi_p, \Psi_{p_j})}{\sigma^2}\right) \quad (5.9)$$

où $d(\cdot)$ mesure la distance au carré moyenne entre les parties connues de Ψ_p et un patch valide Ψ_{p_j} , Z_p est une constante de normalisation telle que $\sum_{p_j \in N_s(p)} w_{p,p_j} = 1$, et σ est un hyper-paramètre fixé à $\sigma = 5$. Ce terme de données permet de discriminer les pixels de textures des pixels appartenant à un contour entre deux objets. Les deux pixels peuvent en effet avoir un gradient fort, mais le pixel de contours est distribué de manière plus éparse autour de ses voisins (similarité faible), et peut donc être choisi en priorité.

- Plus récemment, les auteurs de [MARTIN12] ont proposé d'amplifier le terme de données de manière non-linéaire en calculant ces priorités :

$$\hat{D}_p = \exp\left(\frac{D_p}{2\sigma^2}\right) \quad (5.10)$$

avec σ étant un paramètre additionnel. Cette modification renforce la valeur du terme de données en présence d'isophotes très contrastées ($D_p \rightarrow 1$), et l'atténue dans les régions d'images homogènes ($D_p \rightarrow 0$).

Des algorithmes semi-supervisés basés sur les travaux de Criminisi *et al.* ont également été proposés. Par exemple, [PÉRE04] contraint l'espace de recherche des patches en demandant explicitement à l'utilisateur de spécifier des zones de recherche approximatives. L'approche de [SUN05] utilise également des patches localisés le long de courbes spécifiées par l'utilisateur qui traversent la zone masquée. Les résultats de ces approches semi-supervisées deviennent très dépendantes des interactions manuelles demandées à l'utilisateur, et sont donc d'autant plus difficiles à évaluer et impossible à appliquer de manière complètement automatique.

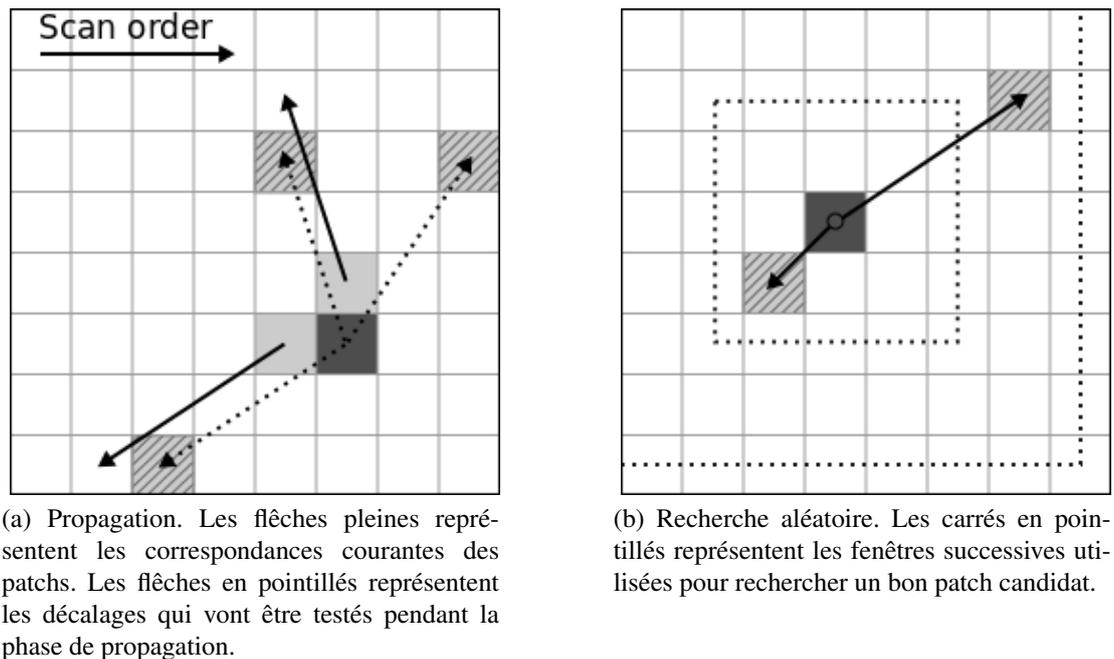
Plusieurs autres approches basées motifs [WONG08, LE MEU11, MARTIN12, GUILLE13] et d'autres approches hybrides [WEXLER07, KAWAI09, NEWSON13] reconstruisent les régions d'image manquantes en mélangeant plusieurs patches candidats. Plus précisément, les K meilleurs patches les plus similaires au patch destination sont trouvés, et la partie manquante est complétée par des patches obtenus par combinaison linéaire de ces K patches. Ce type de synthèse semble pertinent au premier abord, mais produit malheureusement souvent des résultats trop lisses, particulièrement dans les régions texturées.

Recherche de patch sources. Au delà de l'importance de la métrique utilisée pour calculer la similarité entre patches, une composante clé des algorithmes d'inpainting basés motifs (tant en terme de vitesse d'exécution que de qualité visuelle du résultat) est la façon dont on recherche un bon patch source pour la reconstruction.

Tout d'abord, la méthode de recherche pour un algorithme d'inpainting basé patch se doit être la plus efficace possible, *c-à-d.* garantir une qualité optimale de complétion de l'image. Elle doit être également rapide, pour que l'algorithme d'inpainting puisse être effectivement utilisé dans des applications réelles, la recherche des meilleurs patches étant en effet une des étapes les plus coûteuse en temps de calcul. Ce concept de recherche "rapide" va de pair avec la notion de solution approchée, ce qui induit à priori une reconstruction de patches moins bonne qu'avec une recherche complète. Satisfaire à ces deux contraintes de vitesse et de qualité est un des challenges difficiles de l'inpainting. Une méthode simple et relativement efficace de recherche de patches sources est d'utiliser une fenêtre de taille fixe centrée au pixel p à reconstruire. L'espace de recherche étant réduit, on passe en revue beaucoup moins de patch candidats qu'avec une recherche exhaustive sur toute l'image. Une recherche exhaustive est équivalente à la construction d'un *NNF* (*Nearest Neighbor Field*) pour une métrique de similarité donnée.

Une autre solution consiste à approximer le *NNF* avec un *ANNF* (*Approximate Nearest Neighbor Field*) pour la recherche du meilleur patch, qui ne donne pas la solution optimale (*c-à-d* le patch qui maximise la similarité sur toute l'image), mais un patch acceptable. Une *ANNF* entre deux images I_a et I_b se modélise comme une carte d'offsets $\Phi(p)$ qui donne pour chaque point $p \in \mathcal{I}_a$ la localisation relative du centre $q \in \mathcal{I}_b$ du patch Ψ_q qui est (approximativement) le plus similaire à Ψ_p . Pour le cas particulier de l'inpainting, I_a et I_b représentent la même image, restreints à la région connue $\tilde{\Omega}$. Plusieurs méthodes existent pour calculer un *ANNF* :

- Les *KD-trees* sont beaucoup utilisés pour le calcul rapide d'*ANNFs*, notamment pour les algorithmes d'inpainting qui requièrent un grand nombre d'itérations [WEXLER07]. Les *KD-trees* ont été par la suite améliorés [HE12B] pour accélérer le recherche en utilisant une phase de propagation. Notons que ce type de méthode de recherche est également utilisé pour calculer des statistiques rapides sur les offsets de patches dans [HE12A].
- En 2009, Barnes *et al.* ont proposé l'algorithme itératif *PatchMatch* [BARNES09], qui est basé sur trois observations principales :



(a) Propagation. Les flèches pleines représentent les correspondances courantes des patches. Les flèches en pointillés représentent les décalages qui vont être testés pendant la phase de propagation.

(b) Recherche aléatoire. Les carrés en pointillés représentent les fenêtres successives utilisées pour rechercher un bon patch candidat.

FIG. 5.1: Illustration des deux phases principales de l'algorithme *PatchMatch* : propagation (a) et recherche aléatoire (b).

1. La dimensionnalité de l'espace des offsets de patch est beaucoup plus petite que celle de l'espace des patches.
2. Les structures dans les images naturelles sont le plus souvent organisées de manière contigue.
3. On peut espérer qu'avec la loi de grands nombres, une fraction non-négligeable d'un ensemble de localisations aléatoires de patches donnera de bons candidats.

Ainsi, l'idée est d'initialiser un $ANNF \Phi(p)$ de manière complètement aléatoire, puis de raffiner itérativement celui-ci, en alternant deux phases distinctes : une phase de propagation et une phase de recherche aléatoire. La phase de propagation, rappelant ce qu'on trouve dans la méthode de synthèses de texture [ASHIKH01], permet d'améliorer des pans entiers de l' $ANNF$ en recherchant les meilleurs patches candidats à la proximité des offsets utilisé par leurs voisins (Fig 5.1). La seconde phase de *recherche aléatoire* choisit une position aléatoire dans l'image (plus précisément dans une fenêtre centrée de taille décroissante au fur et à mesure des itérations), et teste si ce nouveau décalage tombe sur un meilleur patch candidat. Le but de cette seconde phase est de sortir potentiellement d'un minimum local de l' $ANNF$, et lui assurer une meilleure cohérence globale.

Ces algorithmes sont intéressants en raison de leur capacité à permettre à la fois une cohérence locale et globale lors de la recherche de patch. Par contre, ils ont été initialement imaginés pour travailler sur des images *complètes* (où toute l'information est connue), et qui ne sont pas sujettes à des changements après le calcul de l' $ANNF$. C'est le cas dans diverses problématiques où ils ont été utilisés, comme le débruitage d'images ou la détection d'objets. Mais ces méthodes requièrent des adaptations dans le cas de l'inpainting basé motif, où certaines régions de l'image ont des pixels de valeurs inconnues. Dans ces cas en effet, l' $ANNF$ doit être recalculé au fur et à

mesure que l'image est reconstruite, ce qui implique une phase additionnelle de mise à jour qui doit être la plus rapide possible.

5.2 Quelques contributions à l'état de l'art

La revue technique des méthodes de l'état de l'art que nous avons réalisé dans les sections précédentes nous a permis de mettre le doigt sur certains point-clés des algorithmes d'inpainting basés motifs qui pouvaient être améliorés. A partir de ces quelques observations détaillées ci-dessous, nous proposons des améliorations s'appliquant à chaque étape clé de l'algorithme d'inpainting, et illustrons l'efficacité de ces modifications en comparant les résultats obtenus avec quelques méthodes d'inpainting de l'état de l'art, sur des images difficiles à reconstruire.

5.2.1 Terme de données basé tenseur

Nous avons déjà mentionné que le terme de priorité était l'une des caractéristique les plus importante introduite dans [CRIMIN04] pour sélectionner les endroits où la reconstruction est la plus intéressante pour une itération donnée de l'algorithme d'inpainting. Plus particulièrement, le terme de données D_p (Eq. 5.4) permet de favoriser les points destinations qui sont localisés sur les contours des structures de l'image, orientés le long de la normale au masque d'inpainting. Donner une priorité importante à ces points est effectivement une bonne idée, puisque cela va naturellement favoriser la reconstruction en premier lieu des structures les plus importantes (c-à-d les plus contrastées).

Malheureusement, la formulation originale proposée pour D_p dans [CRIMIN04] a un inconvénient majeur : elle propose que le gradient $\vec{\nabla} I_p$ du point de destination se calcule comme la valeur maximale des gradients de l'image dans le voisinage non-masqué $\mathcal{N}_p \cap I$ de p (Eq. 5.5). En conséquence, quand la taille (fixée) de patch N utilisée pour la reconstruction est grande, le terme de données D_p aura souvent une valeur élevée non seulement pour les pixels du contour à étendre, mais également pour tous les pixels $\in \delta\Omega$ dont la distance à ce contour est inférieure ou égale à N (Eq. 5.5). Cet effet naturel de dilatation du terme de données va avoir tendance à donner trop d'importance à des pixels destination voisins du contour, qui pourront être malencontreusement sélectionnés à la place des pixels effectivement situés sur le contour. C'est le cas par exemple lorsque la forme du masque favorisera localement ces points de non-contours parce que le terme de confiance C_p sera plus élevé en ces points (masque moins courbe en ces points par exemple). Ce fait est illustré en Fig.5.2a sur un cas réel. Le terme de données (Eq. 5.4) est élevé pour tous les points cibles autour de la frontière entre le sable et la mer, et le point cible finalement sélectionné pour la reconstruction *n'est pas centré* sur cette frontière. (Fig.5.2b). Un patch contenant seulement du sable va être alors collé ici, ce qui va complètement casser cette bordure sable-mer.

Comme cela a été discuté en section 5.1, plusieurs termes de données ont déjà été proposés dans la littérature. Mais même s'ils essayent d'améliorer le terme de données initial, ils souffrent tous de quelques inconvénients que nous listons ici :

- Le terme de données proposé dans [LE MEU11] est basé sur le calcul des tenseurs de structure locaux (Eq. 5.7). L'utilisation de ces tenseurs est à la base une excellente idée puisque les tenseurs sont capables de bien modéliser les différentes configurations de va-

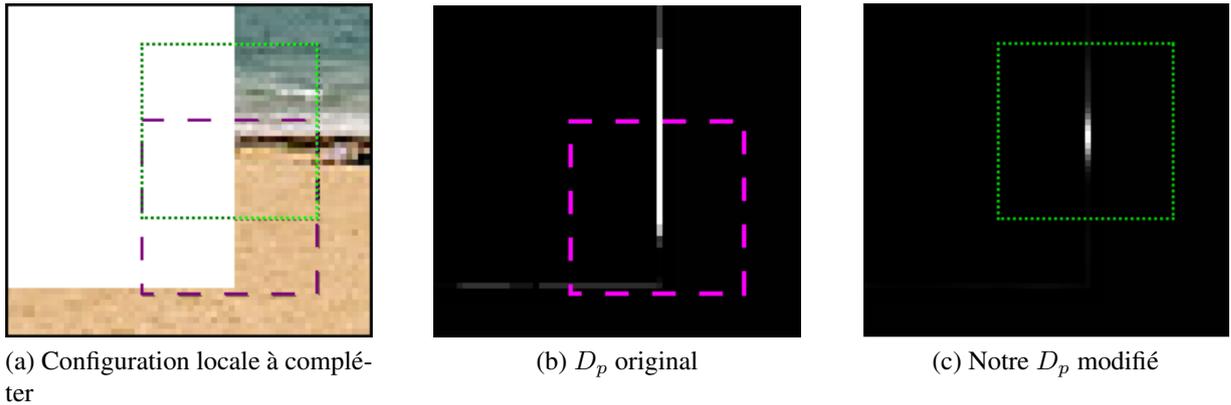


FIG. 5.2: Illustration de l'impact du terme de données. Les carrés violet (b) et vert (c) représentent les patches prioritaires sélectionnés respectivement en utilisant le terme original D_p de [CRIMIN04] (Eq. 5.4) et le terme de données proposé ci-après (Eq. 5.11).

riations locales des images multi-canaux Néanmoins, le terme de donné proposé (Eq. 5.7) repose beaucoup sur le choix d'hyper-paramètres ad-hoc η et α dont le rôle n'est pas clair. Plus embêtant, tel que défini dans [LE MEU11], le terme de données ne prend pas du tout en compte le vecteur normal au masque Ω , ce qui veut dire qu'un contour presque tangent à Ω aura une forte anisotropie qui pourra lui donner une priorité haute pour la reconstruction, alors que cette configuration géométrique n'est pas idéale pour la reconstruction.

- Le terme de données parcimonieux proposé dans [XU10] autorise principalement la bonne gestion des pixels de $\delta\Omega$ appartenants à des textures. Avec le terme de données de Criminisi *et al.* (Eq. 5.4), la plupart des pixels dans ce cas peuvent avoir des gradients forts, donc des priorités de reconstruction fortes également. Comme un patch centré sur un tel pixel est fortement redondant avec les patches de son voisinage, le terme de données proposé par Xu *et al.* (Eq. 5.8) décroît sa priorité. Ceci dit, ce terme de données a deux inconvénients principaux : d'abord, le voisinage d'un pixel p est considéré comme un carré centré en p dont le choix de la taille a un impact important sur ce terme parcimonieux. En réalité, il est très lié à la *taille de la texture* (texton), qui elle n'est pas prédictible. Ensuite, le calcul de ce terme de données a une complexité algorithmique trop importante. A la première itération, pour chaque pixel du contour du masque, on doit calculer de nombreuses distances SSD entre patches, ce qui est *très* coûteux. Et même si cette opération doit être répétée aux itérations suivantes seulement pour quelques pixels (ceux qui ont été complétés aux itérations d'avant), l'ordre de grandeur du temps de calcul global de l'algorithme est une à deux fois supérieure aux algorithmes utilisant des termes de données plus classiques.
- Finalement, la méthode proposée dans [MARTIN12] impose un terme de données suivant une décroissance exponentielle pour le calcul des priorités (Eq. 5.10). La logique derrière cette amplification non-linéaire est de renforcer la valeur du terme de données en présence d'une isophote contrastée ($D_p \rightarrow 1$), sans pour autant l'amplifier dans les régions homogènes ($D_p \rightarrow 0$). Le problème principal de cette heuristique, tout comme pour le terme parcimonieux décrit précédemment, est d'ajouter un paramètre additionnel σ , dont l'influence est énorme et qui doit être *in fine* ajusté manuellement pour chaque image, ce qui est loin d'être pratique.

Pour pallier ces problèmes, nous proposons une approche géométrique simple basée sur l'utilisation des tenseurs de structures [DI ZEN86], utilisés pour modéliser les variations de l'image à l'intérieur d'un patch candidat. Premièrement, cela pour intérêt d'estimer localement les structures dans les images en utilisant une approche multi-canaux corrélée (la corrélation entre les canaux couleurs R,G,B est par exemple prise en compte). Deuxièmement, nous tirons avantage des propriétés algébriques de la somme des tenseurs, pour permettre aux patch cibles contenant plusieurs structures orientées différemment (typiquement des textures) de répondre favorablement, en fonction de l'orientation locale \vec{n}_p du vecteur normal au masque estimé en p . Le terme de données D_p que nous proposons est :

$$D_p = \|\mathbf{G}_p \vec{n}_p\| \quad (5.11)$$

où \mathbf{G} est une moyenne pondérée des tenseurs de structures estimés sur les parties non-masquées du patch cible Ψ_p :

$$\mathbf{G}_p = \sum_{q \in (\mathcal{N}_p \cap (I - \Omega))} w_q \overline{\nabla I}_q \overline{\nabla I}_q^T$$

et w est une fonction 2D gaussienne normalisée, centrée en p . Cette proposition de terme de données peut être interprétée de la façon suivante.

- Quand \mathbf{G} est très anisotrope ($\mathbf{G} \approx \mathbf{u}\mathbf{u}^T$), il y a une direction claire et unique de contour à l'intérieur du patch cible Ψ_p , contour orienté par $\vec{\mathbf{u}}^\perp$. Dans ce cas, notre terme de données (Eq. 5.11) devient approximativement égal à $|\langle \vec{\mathbf{u}}, \vec{n}_p \rangle|$ qui sera grand lorsque le contour sera orienté suivant la normale à Ω . Notons également que D_p sera d'autant plus grand que les points cibles p seront localisés précisément sur le contour de image traversant le patch (car les poids gaussiens w favorisent naturellement le centrage du patch cible sur le contour). Il n'y a plus d'effet de dilatation qui peut causer un choix prioritaire de points cibles non centrés sur un contour effectif de l'image.
- Quand \mathbf{G} est isotrope ($\mathbf{G} \approx \lambda Id$) avec des faibles valeurs ($\lambda \approx 0$), il y a très peu de variations à l'intérieur du patch Ψ_p et p est donc localisé sur une région homogène. Dans ce cas, notre terme de données (Eq. 5.11) a une valeur faible, puisqu'il est approximativement égal à λ .
- Quand \mathbf{G} est isotrope, mais avec des valeurs fortes ($\lambda \gg 0$), le patch cible Ψ_p contient des variations multiples avec des orientations spatiales différentes, caractéristiques de structures contrastées mais complexes comme les textures. Dans ce cas, notre terme de données (Eq. 5.11) aura toujours une valeur forte, quelque soit l'orientation de la normale \vec{n}_p au masque d'inpainting.

L'intérêt d'utiliser ce terme de données en remplacement du terme original est illustré sur la Fig.5.2c où l'on peut voir très clairement une meilleure localisation des points de reconstruction prioritaires, ainsi que la pertinence des patchs cibles sélectionnés à une itération donnée. Comme l'inpainting est un processus itératif glouton où chaque sélection de localisation de reconstruction dépend de ce qui a été fait aux itérations précédentes, nous avons pu observer que l'utilisation de notre terme de données a effectivement un effet énorme sur la qualité finale de l'image reconstruite (Fig.5.3).

5.2.2 Une stratégie de recherche astucieuse

Les méthodes de l'état de l'art inspirées de l'algorithme de Criminisi *et al.* ont beaucoup cherché à améliorer le *terme de priorité* P_p , souvent en modifiant C_p ou D_p , ou en essayant

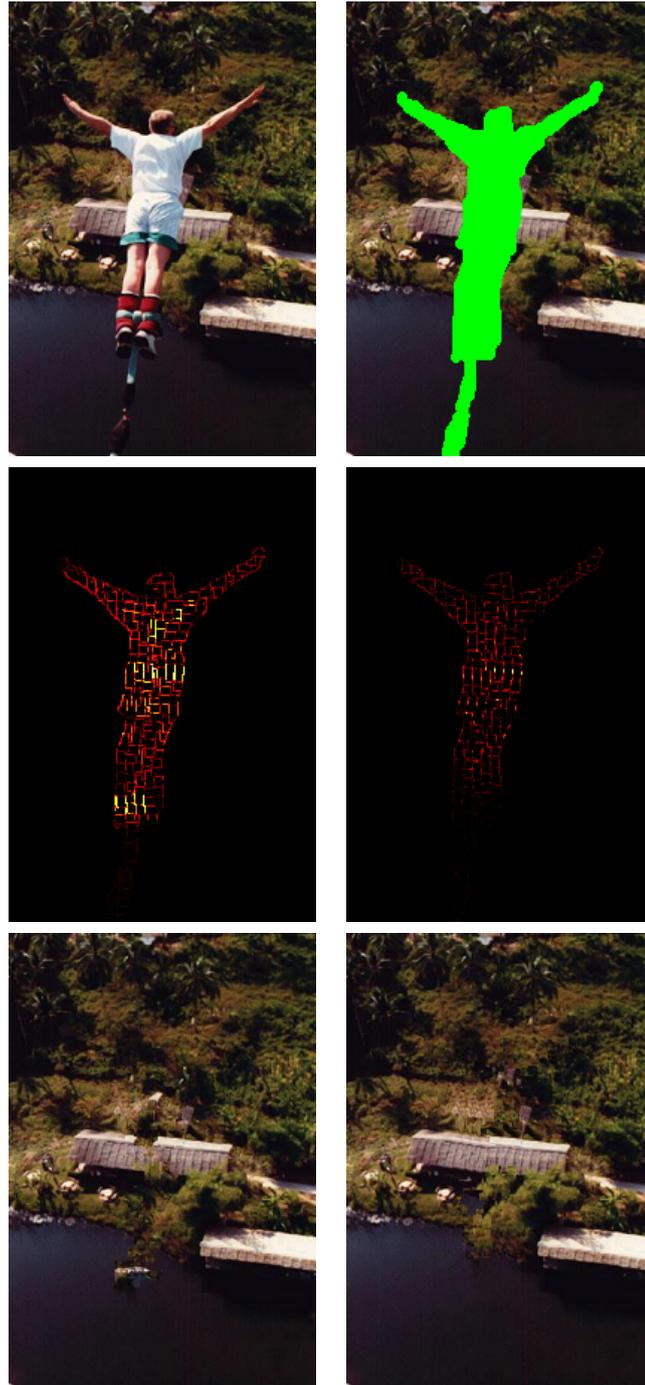


FIG. 5.3: Illustration du *terme de données* proposé pour l'image du saut à l'élastique. En haut : l'image originale, et la partie à reconstruire Milieu : Valeurs du terme de données original de Criminisi *et al.* [CRIMIN04] (à gauche) et du nôtre (à droite) avec une palette de chaleur (rouge foncé pour les valeurs faibles, et jaune pour les valeur élevées). En bas : Résultats d'inpainting associés. A cause de l'effet de dilatation du terme de données original (Eq. 5.4), le toit de l'abri est brisé (à gauche). On peut également remarquer des problèmes de propagation incorrecte de textures d'arbres dans l'eau. Le terme de données proposé (Eq. 5.11) permet de mieux gérer ces structures en évitant des copies de patchs indésirables.

d'améliorer la métrique de similarité entre patchs. La manière de rechercher un bon patch candidat pour la reconstruction est resté un point sous-étudié, alors que c'est une étape clé pour que ce

type d'algorithmes fonctionne correctement. Dans nos expériences, nous avons implémenté plusieurs méthodes différentes de recherche de patches candidats : une fenêtre de recherche simple, une adaptation de l'algorithme *PatchMatch*, ou encore le calcul de statistiques sur des offsets de déplacements.

L'utilisation d'une fenêtre de recherche simple peut être rapide, mais devient rapidement compliquée à gérer dans le cas de grands trous à combler. De plus, la proximité spatiale des échantillons parcourus avec le patch cible fait qu'une cohérence globale de reconstruction n'est pas toujours assurée. L'algorithme *PatchMatch* requiert plusieurs adaptations pour être utilisé dans le contexte des méthodes d'inpainting gloutonnes de type Criminisi. Le critère SSD doit en effet prendre en compte l'aspect non-complet des patches à comparer (certaines valeurs sont inconnues), et la mise à jour d'un *ANNF* doit donc être réalisée à chaque fois qu'un patch cible est partiellement reconstruit. Cette dernière contrainte fait que l'algorithme *PatchMatch* modifié devient beaucoup lent que la version originale décrite dans [BARNES09]. Enfin, l'utilisation de statistiques locales d'offsets n'est pas non plus aisée, car le nombre de pics des histogrammes sélectionnés devient trop faible, ce qui amène à des inconsistences locales de reconstruction. Notre contribution cherche ici à combiner la rapidité d'une fenêtre de recherche, et la cohérence géométrique globale propre à *PatchMatch* et aux statistiques sur les offsets de déplacements. L'idée est de rechercher les patches candidats à l'intérieur d'une fenêtre de recherche, mais de se rappeler pour chaque patch cible l'offset de déplacement qui a été utilisé, pour les réutiliser comme guides dans les itérations suivantes de l'algorithme. De cette manière, nous construisons un ensemble de sites de recherches où nous pouvons réaliser plusieurs recherches de patches rapidement (les fenêtres étant de taille réduite).

5.2.2.1 Analyse des méthodes de recherche de patches de l'état de l'art

Fenêtre de recherche.

Une fenêtre de recherche définit une zone rectangulaire plus petite que l'image entière, où un bon patch candidat est recherché pendant le processus d'inpainting. De cette façon, seuls quelques patches sont analysés durant cette phase de recherche, et la reconstruction est donc considérablement accélérée puisque seulement un petit nombre de *SSDs* sont calculés. Néanmoins, l'utilisation d'une fenêtre de recherche peut être problématique. D'abord parce que le fait que la taille de la fenêtre soit réduite risque de mener à des reconstructions pauvres en détails : les mêmes patches de textures risquent par exemple très fortement d'être recollés dans le résultat final. Ensuite, parce que même avec une taille de fenêtre suffisante, on se retrouve à analyser et potentiellement choisir des patches candidats qui ne sont pas localement cohérents avec la géométrie que l'on cherche à reconstruire.

Un des problèmes principaux de ce type de schémas de recherche est que les patches recherchés le sont uniquement dans la zone connue $\mathcal{I} \setminus \Omega$ de l'image, pour éviter la copie de pixels qui ont déjà été reconstruits aux itérations précédentes. Et quand la taille du masque d'inpainting est plus grande que la taille de la fenêtre de recherche, tout se passe bien lors des premières itérations (Fig. 5.4a), mais le processus se gâte lorsque l'espace de recherche devient trop contraint pour permettre l'obtention d'un bon patch candidat pour la reconstruction (Fig. 5.4b). Une solution ad-hoc [BORNAR02] consiste à agrandir la taille de la fenêtre de recherche de manière graduelle au fur et à mesure des itérations. Mais même si l'utilisation d'une fenêtre de recherche réduite induit une meilleure cohérence locale de la géométrie reconstruite, elle limite aussi le choix possible des paramètres de l'algorithme et ne donne aucune garantie de cohérence globale.

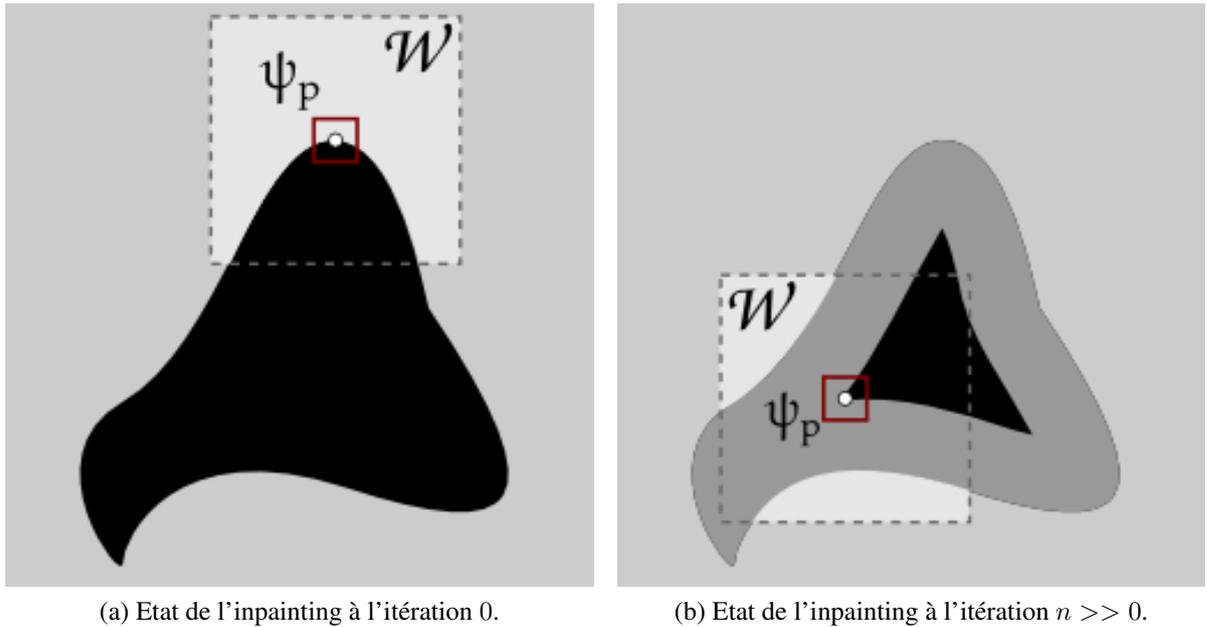


FIG. 5.4: Illustration du problème de la taille de la fenêtre de recherche. La quantité de patches à analyser est beaucoup moins importante, dans la fenêtre de recherche à l'itération n .

L'algorithme *PatchMatch*. L'algorithme *PatchMatch* [BARNES09] a été initialement pensé comme un algorithme de calcul de correspondances entre les patches d'une image avec les patches d'une autre image. Pour cette raison, l'utilisation de cet algorithme pour le problème de l'inpainting basé motifs nécessite quelques adaptations. Dans les méthodes de l'état de l'art utilisant *PatchMatch*, la façon dont est gérée les données manquantes reste peu analysée, malgré l'importance de ce processus. La façon la plus évidente de faire et de modifier la façon dont on compare deux patches. Dans le cas de l'inpainting basé motifs, l'Eq. 5.1 peut par exemple être utilisée.

Une seconde adaptation nécessaire est liée au fait que *PatchMatch* est généralement appliqué sur des images fixes. Le fait que le contenu de l'image change au fur et à mesure des itérations d'inpainting nécessite potentiellement la mise à jour des $\Phi(p)$ pour tous les centres des patches dont le contenu a changé, c-à-d $\{q \mid \mathcal{N}_q \cap \mathcal{N}_p \neq \emptyset\}$. En pratique, cette phase de mise à jour est très coûteuse en temps de calcul, et fait de *PatchMatch* un algorithme difficile à mettre en pratique dans le cas des algorithmes d'inpainting basés motifs **incomplets** avec une approche gloutonne.

Statistiques sur les offsets. Une idée intéressante avec les *ANNF* est de bénéficier de calcul statistiques sur les offsets de déplacements [HE12A], et ceci peut être relativement facile à adapter aux algorithmes d'inpainting basé motifs. Nous avons mené des expériences en ce sens, où une phase de pré-inpainting sert de base pour le calcul d'un *ANNF* (utilisant *PatchMatch*).

Les K offsets dominants peuvent être extraits ensuite pour guider la recherche des patch candidats lors d'une deuxième passe d'inpainting. Cette technique fonctionne bien dans la plupart des cas, mais nous avons remarqué que l'utilisation de K offsets dominants met quelquefois de côté certaines configurations géométriques, ce qui amène à des reconstructions locales non naturelles. On peut penser également à créer artificiellement quelques offsets en demandant à l'utilisateur de tracer des traits de guide dans les images à inpainter (Fig. 18 de [HE12A]) pour

résoudre ce problème. Mais dans ce cas, la méthode devient supervisée, et sort du cadre de l'inpainting complètement automatique d'images qui était notre but premier.

5.2.2.2 Contribution à une recherche de patches rapide et perspicace

La contribution que nous proposons ici regroupe plusieurs idées similaires utilisées dans les algorithmes de recherche décrits en section précédente. La stratégie proposée consiste à détecter dans un premier temps les patches cibles déjà reconstruits $\Psi(p)$, et d'utiliser leurs offsets pour définir plusieurs *sites* de recherche susceptibles de contenir un bon patch candidat pour la reconstruction (Fig. 5.5). La zone de recherche est donc composée de multitudes de petites fenêtres de recherche, chacune centrée par rapport aux coordonnées des patches sources précédemment utilisés. En pratique, l'ensemble des patches voisins reconstruits $\{\Psi_c\}$ d'un patch cible Ψ_t contient les patches pour lequel $\mathcal{N}_c \cap \mathcal{N}_t \neq \emptyset$.

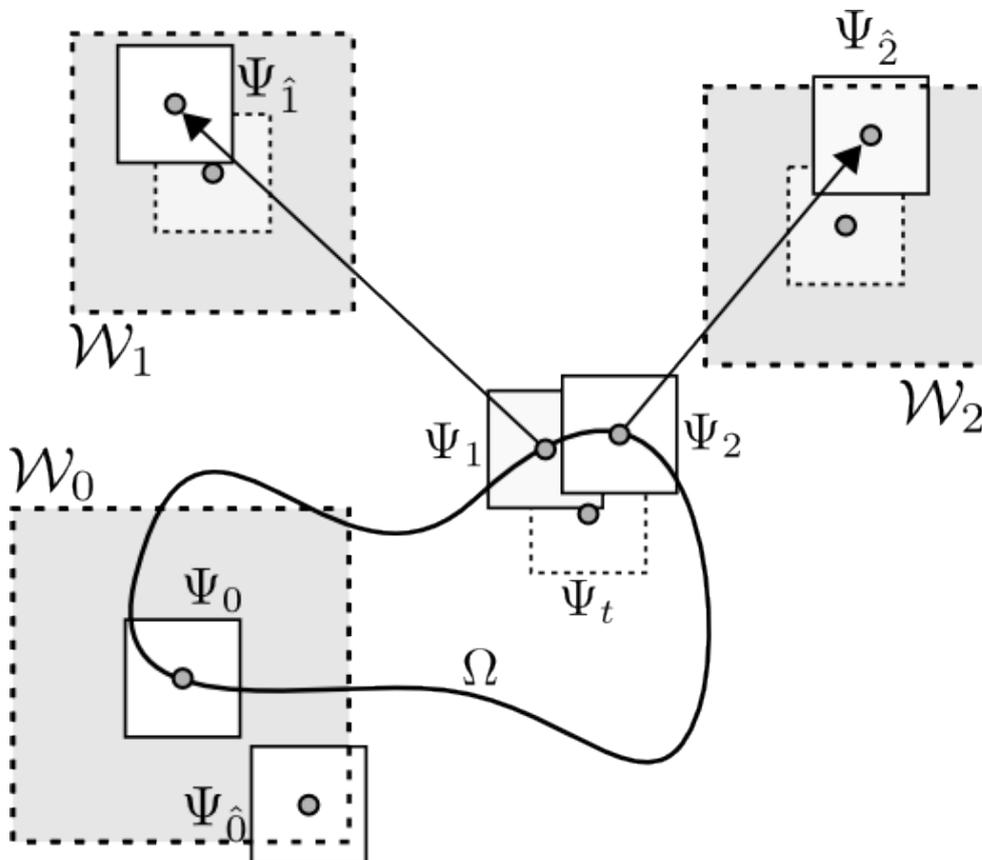


FIG. 5.5: Illustration de notre méthode rapide de recherche pour les algorithmes d'inpainting basés motifs. Chaque patch reconstruit Ψ_1, Ψ_2 fournit la localisation d'un site $\mathcal{W}_1, \mathcal{W}_2$ aidant à trouver un bon patch candidat pour reconstruire Ψ_t . Notons qu'un petit site de recherche \mathcal{W}_t centré en Ψ_t est également utilisé (non dessiné sur la figure). Comme Ψ_0 n'a pas de patches proches déjà reconstruits, la zone de recherche associée est simplement une fenêtre de recherche classique \mathcal{W}_0 centrée en celui-ci.

Pour chaque patch Ψ_p que l'on cherche à reconstruire, la recherche d'un bon patch candidat est effectuée de la façon suivante. D'abord, on récupère l'ensemble des offsets de déplacements

$\Phi(p)$ des patches voisins qui ont déjà été reconstruits aux itérations précédentes. La taille des fenêtres de recherche $\mathcal{W}(p)$ qui seront ensuite utilisées est calculée de la façon suivante :

$$w_{size}(p) = \begin{cases} \gamma & \text{if } |\Phi(p)| = 1 \\ \gamma\alpha\sqrt{|\Phi(p)|} & \text{sinon} \end{cases} \quad (5.12)$$

où γ est la taille maximale autorisée pour une fenêtre de recherche, et $\alpha \in [0, 1]$ est un facteur d'échelle donnant la quantité d'espace de recherche à accorder à chaque sous-site : plus le nombre de sites est grand, plus leur fenêtre de recherche associée est petite. Une fois que ces sites ont été définis, une recherche classique par fenêtre est réalisée dans chacune de ces multiples sous-fenêtres :

$$\Psi_{\hat{p}} = \left\{ \Psi_q \mid q = \arg \min_{\bigcup_{\hat{q} \in \Phi(p)} \mathcal{W}_{\hat{q}}} d(\Psi_p, \Psi_{\hat{q}}) \right\}$$

Les avantages de cette méthode de recherche sont multiples :

- Comme les sous-fenêtres de recherche se chevauchent souvent en pratique (même si ce n'est pas systématique), le nombre de patches candidats à examiner est toujours inférieur au cas d'une unique fenêtre de recherche [CRIMIN04]. Cela accélère grandement la rapidité de la phase de recherche.
- Le problème de la taille de la fenêtre de recherche, illustré en Fig. 5.4 ne se produit plus, puisque des sites de recherche peuvent être toujours trouvés à l'extérieur du masque d'inpainting.
- Comme moins de patches candidats sont examinés, le *ANNF* calculé peut ne pas être optimal (au sens de la *SSD*), mais cela a souvent pour effet de générer de meilleurs résultats visuels. les sous-fenêtres de recherche sont en effet centrées à des positions qui ont une cohérence géométrique forte vis à vis des complétions de patches réalisées aux itérations précédentes. En conséquence de quoi, les patches recollés assurent en général une meilleure cohérence géométrique locale, même si leur *SSD* est plus élevée.
- Notre stratégie de recherche, comme celle proposée par Ashikhmin pour la synthèse de textures dans [ASHIKH01], parvient à assurer une meilleur cohérence globale, en cherchant uniquement des patches à des positions pertinentes, et on devient de fait moins sensible au paramètre de la taille des patches utilisés. On peut en effet se retrouver à copier/coller des morceaux entiers de larges structures et textures, avec des patches dont la taille est inférieure à la taille de ces structures.

La Fig.5.6 illustre la pertinence de ce schéma de recherche de patches et le compare avec la recherche classique par fenêtre unique, pour une image difficile qui contient à la fois des macro-structures et des macro-textures. Bien que les résultats d'inpainting semblent à première vue similaires, notons que notre méthode nécessite seulement l'utilisation de patches 17×17 (en bas à gauche), quand des patches 23×23 sont nécessaires avec une méthode de recherche de patches plus classique (en bas à droite). Si on se limite à des patches 17×17 avec la méthode classique, on dégrade de manière visible le résultat de la reconstruction (en haut à droite).

Ainsi, en mélangeant des idées provenant de plusieurs algorithmes de l'état de l'art en inpainting et synthèse de textures, nous avons pu élaborer un algorithme de recherche de patch original et efficace capable de reconstruire des parties manquantes d'images en maintenant une cohérence géométrique à la fois locale et globale, et ceci en améliorant en plus les temps de calcul des approches classiques. Malgré tout, il apparaît quelquefois des artéfacts visuels de reconstruction dûs au recollement "sauvage" de patches les uns à côtés des autres, comme cela se



(a) Image à compléter

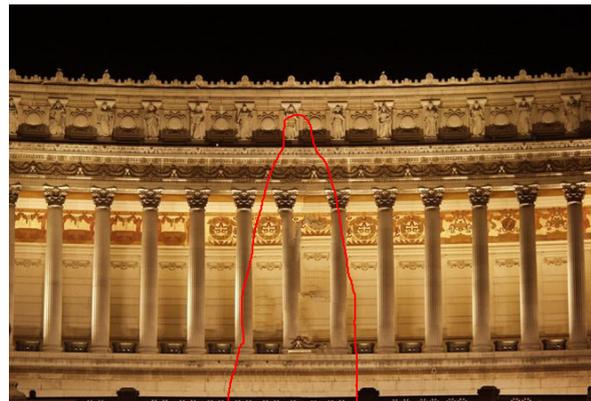
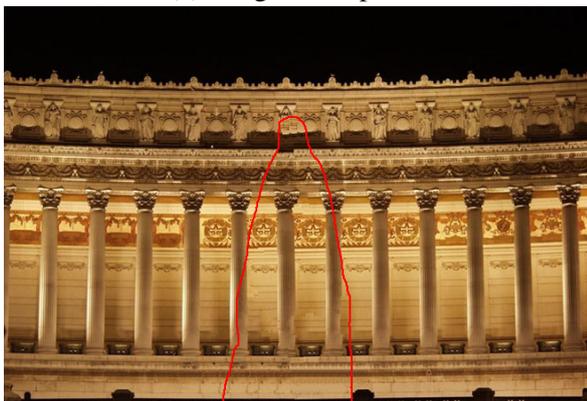
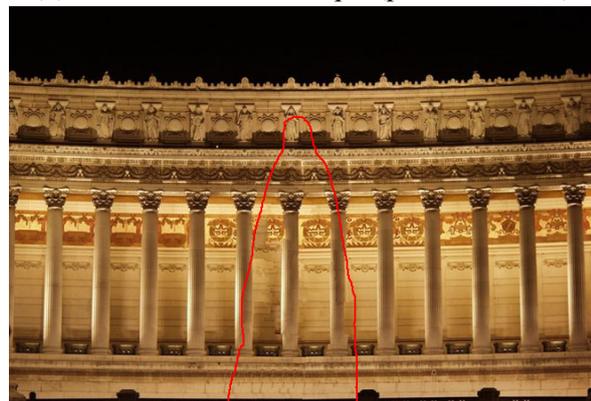
(b) Avec recherche classique (patches 17×17)(c) Avec notre recherche (patches 17×17)(d) Avec recherche classique (patches 23×23)

FIG. 5.6: Comparaisons de résultats d'inpainting entre la recherche classique utilisant une unique fenêtre (colonne de droite) et le schéma de recherche que nous proposons (en bas à gauche). Les meilleurs résultats ont été obtenus avec des patches 23×23 pour la recherche classique (en bas à droite) et avec des patches 17×17 pour notre algorithme de recherche (en bas à gauche). La taille de la fenêtre de recherche initiale a été fixée à 400×400 . L'inpainting utilisant une recherche de patch classique et des patches 17×17 (en haut à droite) dégrade trop la structure des colonnes reconstruites.

passé également avec toutes les méthodes gloutonnes d'inpainting basées motifs. Dans la section suivante, nous nous proposons donc d'attaquer ce problème, en proposant une solution efficace de réduction d'artéfacts par mélange spatial de patches.

5.2.3 Mélange spatial de patches

Les algorithmes d'inpainting basés motifs du style de Criminisi *et al.* génèrent assez souvent des artéfacts visuels de type “effets de bloc” lors de la reconstruction. Cela est principalement causé par le fait que l'on copie/colle des morceaux de patches les uns à côtés des autres, qui quelquefois ne sont pas assortis aux niveaux des frontières du recollage. Et ceci arrive même en utilisant un schéma de recherche de meilleur patch perspicace, comme celui présenté précédemment. Ces artéfacts étant relativement classiques, plusieurs méthodes de la littérature ont tenté des pistes pour les réduire. Dans [LE MEU11], un moyennage des K meilleurs patches candidats est utilisé pour remplir le patch cible, mais selon le nombre de patches K considéré, le résultat peut facilement être sur-lissé (si K est grand) ou au contraire sous lissé (si K est petit). Le type

de reconstruction pixel à pixel utilisé dans [WEXLER07] fournit une manière implicite de ne pas créer trop d'artéfacts de blocs, mais là encore le résultat peut sembler un peu trop régularisé, où présente des déformations de textures. En s'inspirant de la technique du mélange basé Poisson [PÉRE03], utilisé dans l'algorithme d'inpainting [PÉRE04], nous proposons une méthode *rapide et efficace* pour réduire ces effets de blocs lors de la reconstruction, basée sur un mélange spatial de patches voisins, là où les artéfacts visuels ont le plus de chance d'être présents. Contrairement à [PÉRE04] où les interfaces entre blocs sont connus à l'avance (car les patches sont copiés entièrement), nous avons ici une première phase de détection des artéfacts visuels à partir de l'analyse du résultat de l'algorithme d'inpainting "classique". Dans un second temps, l'analyse des offsets de patches utilisés pour la reconstruction nous permet d'appliquer un mélange spatial de patches astucieux, pour réduire de manière importante la visibilité de ces artéfacts de recollage.

5.2.3.1 Détection des artéfacts

Les résultats de reconstruction des méthodes classiques d'inpainting basés motifs ne présentent pas des artéfacts de blocs partout dans l'image résultat, et il n'est donc pas souhaitable de modifier le résultat là où le recollage est fait de manière pertinente. Nous proposons donc une méthode de détection des artéfacts visuels de blocs, basée sur ces deux hypothèses définissant une localisation probable d'un tel artéfact : 1) La variation de luminosité ou de couleur est importante là où l'artéfact est localisé (ce qui indique une discontinuité dans la géométrie reconstruite), et 2) Les patches collés les uns à côté des autres à ces endroits proviennent d'endroits très différents dans l'image, et ont de grandes chances d'être de natures différentes (pas de prolongement d'une texture par exemple). Le schéma de détection (Fig. 5.7) est donc divisé en deux parties : le calcul d'un *champ de rupture*, suivi d'une extraction des *points de rupture*.

Calcul du champ de rupture. Le champ de rupture associe, à chaque point de l'image reconstruite, un poids indiquant l'amplitude de l'artéfact visuel potentiellement présent. Il est défini à partir des deux hypothèses précédentes, de la façon suivante :

$$R(p) = \frac{|\nabla I_p| \cdot |\nabla \Phi(p)|}{\alpha} \text{ avec } R(p) \in [0, 1] \quad (5.13)$$

où $\alpha = \max_{q \in \mathcal{I}} (|\nabla I_q| \times |\nabla \phi(q)|)$ est un facteur de normalisation. Cette équation donne des valeurs fortes à la fois aux endroits où les variations locales sont importantes (grandes valeurs de $|\nabla I_p|$) et où les patches ayant servi à la reconstruction proviennent de localisations sources différentes (grandes valeurs de $|\nabla \Phi(p)|$). À l'inverse, lorsque l'une des deux hypothèses est vérifiée, les valeurs du champ de rupture deviennent localement faibles.

Points de rupture. Dans l'optique d'améliorer le résultat d'inpainting seulement là où c'est nécessaire, nous extrayons un ensemble de *points de rupture* \mathcal{E} à partir des maxima du champ de rupture R , simplement en le seuillant suivant une valeur de seuil τ . Plus τ est élevé, moins on réalisera de corrections locales par mélange spatial de patches. À noter que ce paramètre est facile à régler, puisque le facteur de normalisation α de l'équation 5.13 fait que les valeurs de τ sont compris dans $[0, 1]$.

5.2.3.2 Algorithme de mélange spatial de patches

Le principe du lissage spatial de patches est de fusionner ensemble les informations provenant de patches voisins qui se recouvrent de manière à ce que les discontinuités de recollage



(a) Résultat d'inpainting basé motifs. (b) Carte estimée des amplitudes locales d'artéfacts visuels. (c) Résultat d'inpainting basé motifs + application de notre algorithme de mélange spatial de patches.

FIG. 5.7: Séquence de détection d'artéfacts visuels de blocs, à partir du résultat d'inpainting original (à gauche), jusqu'à l'image corrigée par mélange spatial de patches (à droite).

de ces patches deviennent moins visibles. Pour chaque point $p \in \Omega$, le mélange spatial est effectué par simple combinaison linéaire de tous les pixels recouvrant p , provenant de l'ensemble de tous les patches $\{\Psi_1, \dots, \Psi_n\}$ ayant été utilisés localement pour la reconstruction. Nous détaillons ci-dessous notre méthode proposée de mélange spatial de patches *guidé par des tenseurs*, qui a l'avantage de supprimer les incohérences géométriques locales de reconstruction tout en préservant les structures significatives et les textures présentes dans les images reconstruites.

Modèle tensoriel pour le mélange spatial de patches.

L'idée du modèle de mélange que nous proposons ici est d'avoir le comportement suivant : (1) Le mélange dans une région homogène doit être réalisé avec une forte amplitude, et de manière isotrope, (2) Le mélange dans une région texturée doit être réalisé avec une amplitude plus faible, mais également de manière isotrope, et (3) le mélange effectué sur un contour doit être réalisé avec une amplitude également faible, et de manière anisotrope pour prendre en compte l'orientation principale de la structure.

La première étape est donc de construire un champ de tenseurs \mathbf{B} (que nous appelons *tenseurs de mélange*) qui répond à ces contraintes. Les valeurs propres $\lambda_{\mathbf{B}\{1,2\}}$ et vecteurs propres $\mathbf{e}_{\mathbf{B}\{1,2\}}$ des tenseurs de mélange $\lambda_{\mathbf{B}}$ sont choisis pour représenter respectivement les amplitudes/largeurs et les orientations avec lesquelles le mélange doit se réaliser. Comme on sait que les tenseurs de structure \mathbf{S} donnent de bonnes informations sur la géométrie locale des structures en chaque pixel de l'image, nous les utilisons comme base pour la construction de notre modèle tensoriel de mélange.

Cette construction se réalise en trois étapes :

- Les valeurs propres normalisées des tenseurs de structures sont calculées :

$$\hat{\lambda}_{\mathbf{S}(p)i} = \frac{\lambda_{\mathbf{S}(p)i}}{\max_{p \in \mathcal{I}} \lambda_{\mathbf{S}(p)i}} \quad (5.14)$$

- Les largeurs de mélange (les valeurs propres de tenseurs de mélange) sont définies à partir des valeurs propres des tenseurs de structure (inspiré par nos travaux précédents [TSCHUM05]) :

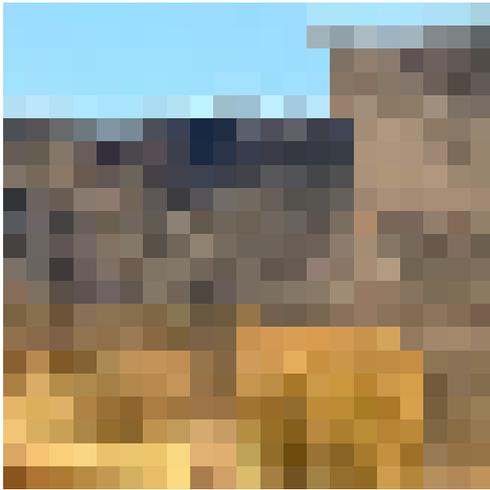
$$\lambda_{\mathbf{B}i} = \frac{1}{(1 + \hat{\lambda}_{\mathbf{S}1} + \hat{\lambda}_{\mathbf{S}2})^{\gamma_i}} \quad (5.15)$$

où γ_i ($i \in \{1, 2\}$) sont des paramètres fixés contrôlant l'anisotropie globale des tenseurs. Des exemples de choix différents de γ_i sont illustrés en Fig. 5.8.

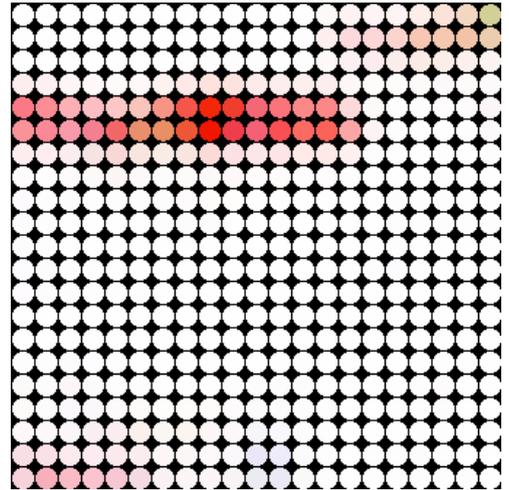
– Finalement, on construit les tenseurs de mélange pour chaque pixel de la façon suivante :

$$\mathbf{B} = \lambda_{\sigma\mathbf{B}1} \mathbf{e}_{\mathbf{S}1}^{\perp T} \mathbf{e}_{\mathbf{S}1}^{\perp} + \lambda_{\sigma\mathbf{B}2} \mathbf{e}_{\mathbf{S}2}^{\perp T} \mathbf{e}_{\mathbf{S}2}^{\perp} \quad (5.16)$$

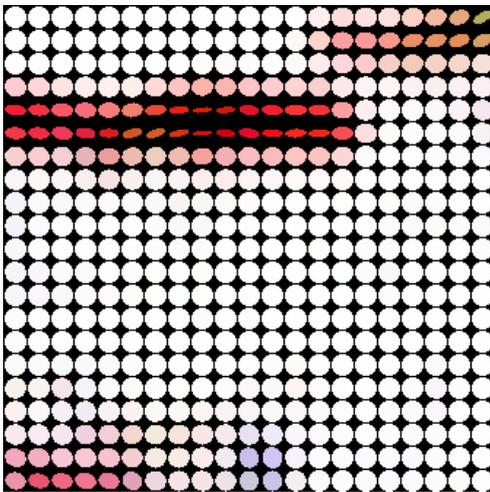
où $\lambda_{\sigma\mathbf{B}i} = \sigma_{\mathbf{B}} \lambda_{\mathbf{B}i}$, sont les valeurs propres finales choisies pour les tenseurs de mélange.



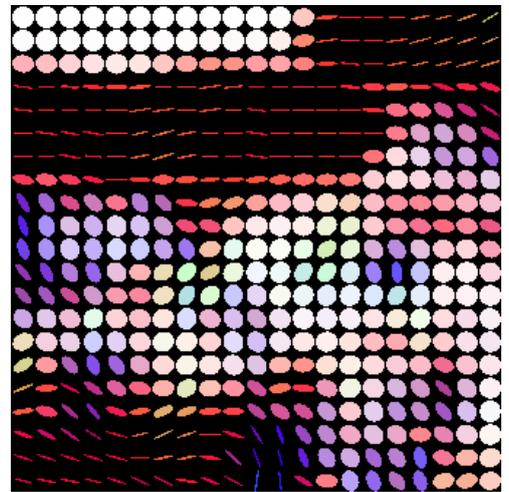
(a) Sample image.



(b) $\frac{\gamma_2}{\gamma_1} \approx 1$



(c) $\frac{\gamma_2}{\gamma_1} > 1$



(d) $\frac{\gamma_2}{\gamma_1} \rightarrow \infty$

FIG. 5.8: Influence du choix des paramètres γ_i sur le champ tensoriel obtenu \mathbf{B} .

Calcul du mélange guidé par les tenseurs de mélange. L'application du processus de mélange spatial de patches permet de générer une nouvelle image pixel à pixel, en étant dirigé par le champ tensoriel de mélange $\mathbf{B}(p)$. L'approche proposée repose sur le calcul d'une combinaison linéaire de pixels provenant de plusieurs patches qui se recouvrent entre eux. Pour chaque point $p \in \Omega$, et pour chaque composante k de l'image multi-valuée, le résultat du mélange est calculé de la façon suivante :

$$J^k(p) = \frac{\sum_{\psi_q \in \Psi_p} w_{\mathbf{B}(p,q)} \psi_q^k(p-q)}{\varepsilon + \sum_{\psi_q \in \Psi_p} w_{\mathbf{B}(p,q)}} \quad (5.17)$$

où $\Psi_p = \{\psi_0, \dots, \psi_n\}$ représente l'ensemble des patches sources recouvrant p , et w_B est une fonction de poids gaussiens anisotropes définis par :

$$w_B(p, q) = e^{-\frac{X^T B(p)^{-1} X}{2\sigma_B^2}} \quad \text{with} \quad X = q - p \quad (5.18)$$

L'utilisation de cette expression pour le mélange a pour effet de mélanger spatialement les patches de manière importante dans les zones homogènes, et de manière plus subtile pour les zones de textures ou contenant des contours de structures. Dans le premier cas, on élimine donc les discontinuités dues au recollement de patches, alors que dans le second cas, on préserve la finesse des détails des structures ou textures synthétisées.

Rendre le mélange plus rapide. Cette formulation pour le mélange spatial de patch est en pratique assez lente à calculer car les tenseurs doivent être définis et considérés en chaque pixel du masque. Fort heureusement, il existe une manière simple permettant d'accélérer ce calcul de mélange de manière approchée, en ne considérant qu'un seul tenseur de mélange au centre des patches pour le calcul. Avec cette formulation rapide, on ne doit donc calculer plus qu'un seul poids de mélange par patch. En pratique, cette approximation ne dégrade pas le résultat visuel du mélange spatial, et rend le mélange suffisamment rapide pour devenir une étape négligeable par rapport au temps passé à l'inpainting basé motif proprement dit. Le paramètre principal de notre algorithme de mélange est l'amplitude du mélange, qui est facile à régler, puisque il est principalement relié à la taille des patches utilisés pour la reconstruction. De plus, l'algorithme de mélange étant très rapide à calculer, et déconnecté du processus d'inpainting proprement dit, il est tout à fait envisageable de rendre ce paramètre réglable par un utilisateur en temps quasi-réel.

La Fig. 5.9 montre une image reconstruite par inpainting, et compare les résultats obtenus après mélange spatial de patches pour deux types de champs de tenseurs différents, l'un étant partout isotrope, l'autre anisotrope (différents réglage de paramètres γ_i dans l'Eq. 5.15). On peut en particulier remarquer que l'utilisation de tenseurs anisotropes permet de mieux préserver les détails des structures et des textures lors du mélange.

5.3 Résultats sur des images couleurs

La Fig. 5.10 illustre les résultats de la méthode d'inpainting finale que nous proposons pour les images 2D (incluant toutes les améliorations proposées dans ce chapitre). Les images traitées sont des images naturelles qui contiennent à la fois des structures et des textures complexes. Nous avons comparé ces résultats avec d'autres méthodes de l'état de l'art, extraits de leurs articles respectifs (quand l'expérience était disponible), ou recalculées lorsque le code était disponible. Comme on peut le voir, ces comparaisons qualitatives montrent que les résultats que nous avons obtenus sont visuellement équivalents ou meilleurs, sur ces images difficiles.

Extension au traitement de séquences vidéos. Nous avons également pu étendre de manière assez directe nos améliorations d'inpainting pour le cas des séquences vidéos. L'extension de notre méthode à l'inpainting vidéo est faite en considérant que l'ensemble des *frames* d'une vidéo forment une image volumétrique 3D (Fig.5.11), et en modifiant l'algorithme d'inpainting pour lui faire prendre en compte des patches cubiques (plutôt que des patches carrés). Cela affecte notamment l'étape de recherche du meilleur patch, ainsi que l'étape de mélange spatial de patch, dont le comportement géométrique local se base sur un champ de tenseurs 3×3 plutôt que 2×2 (Fig.5.12). Ces modifications sont naturelles, et permettent de bien prendre en

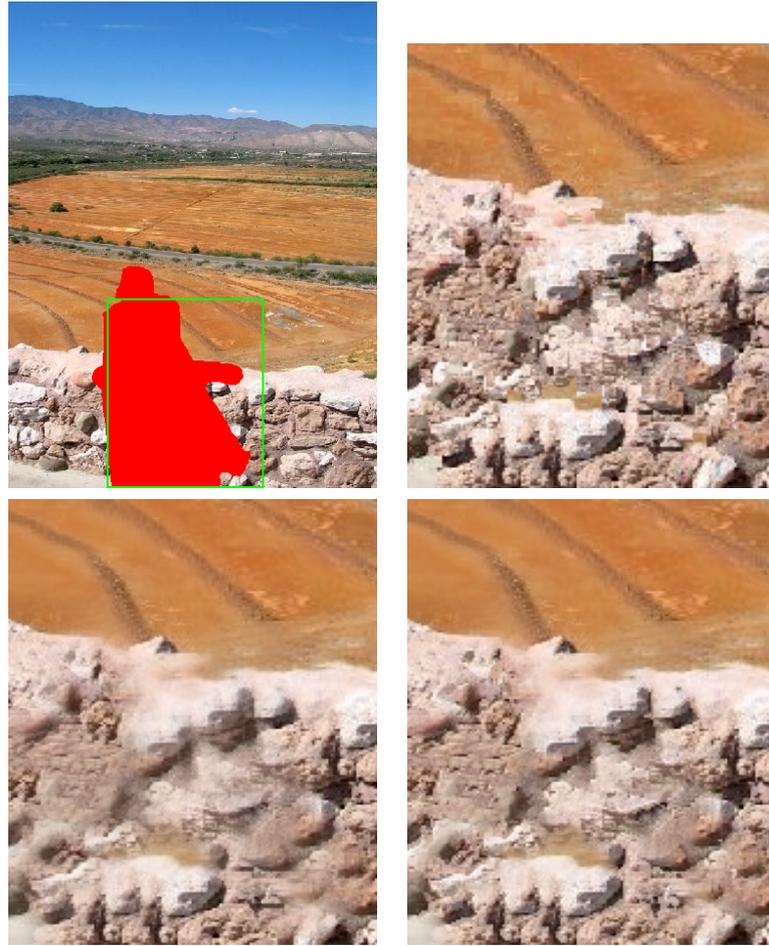


FIG. 5.9: Comparaison de résultats d'inpainting, sans mélange (en haut à droite), avec mélange purement isotrope (en bas à gauche), et mélange anisotrope (en bas à droite).

compte les mouvements des objets dans la séquence à compléter, contrairement à une approche de reconstruction qui serait réalisée sur chaque *frame* séparément.

5.4 Inpainting de paires stéréoscopiques avec carte de profondeur

La télévision 3D et le cadre plus général du rendu indépendant du point de vue (*Free-Viewpoint Rendering*) sont devenus des technologies prometteuses pour l'émergence de nouvelles expériences multimédias (telles que le cinéma 3D, la diffusion ou l'affichage de contenus 3D). Le rendu basé profondeur (*Depth Image Based Rendering*) est ainsi devenu un point important lors de la capture vidéo/photo d'une scène : cela consiste à récupérer, en plus de l'information photométrique classique, l'information de profondeur de la scène. Une carte de profondeur encode pour chaque pixel/élément sa profondeur relative à l'origine de la caméra. A l'aide de cette paire d'images, il est alors possible de synthétiser une vue virtuelle de la scène comme si celle-ci avait été capturée d'un point de vue différent : ce processus de transformation est plus connu sous le nom de *warping* [FEHN04].

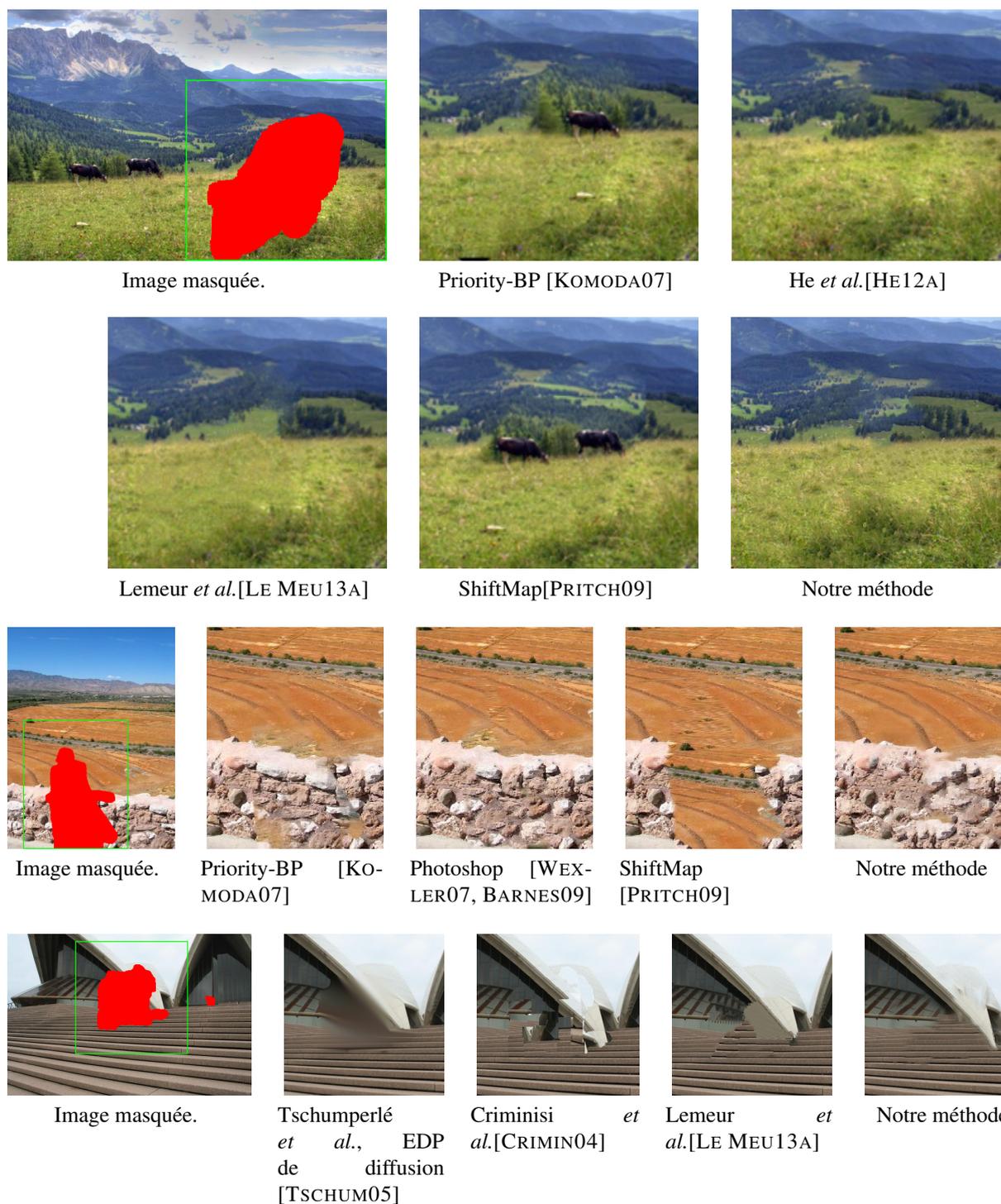


FIG. 5.10: Comparaisons de résultats d'inpainting avec différentes méthodes de l'état de l'art, sur des images naturelles difficiles : *vache*, *mur*, et *opéra de Sydney*.

Un problème de taille se pose cependant lorsqu'une telle vue virtuelle est synthétisée : étant donné que les objets ne sont pas translattés de la même manière selon leur profondeur relative, il arrive que des zones de l'arrière-plan (*background*) jusqu'alors cachées par un objet ayant une profondeur moindre (*foreground*) dans la vue originale aient à être rendus dans la vue synthéti-

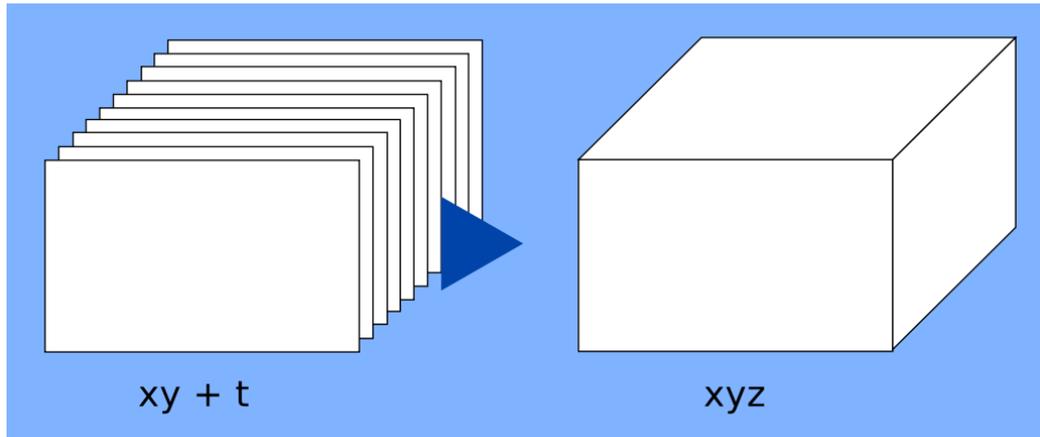


FIG. 5.11: Séquence vidéo, vue comme une image volumique

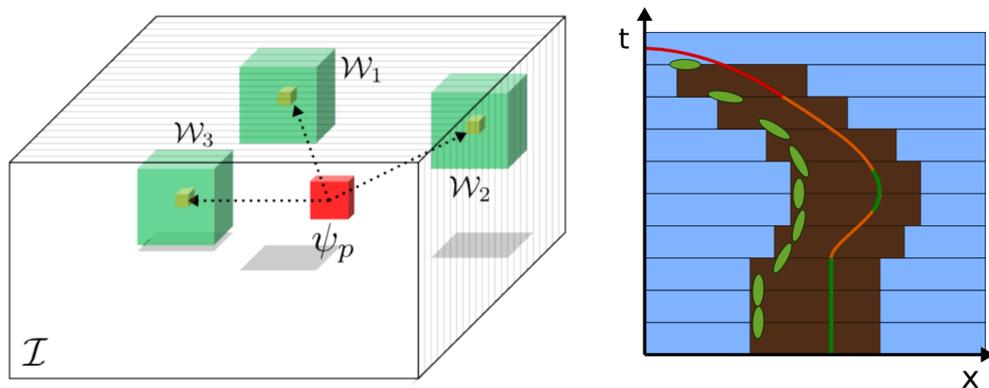


FIG. 5.12: Adaptations nécessaires pour la recherche de meilleurs patches (à gauche) et de mélange spatial de patch (à droite), afin de réaliser l'inpainting vidéo.

sée. Ces parties inconnues sont appelées occultations, et le processus permettant de les restaurer est appelé désoccultation (qui est un cas particulier du problème plus général d'inpainting). Il est important de noter ici que ces occultations apparaissent à la fois dans l'image couleur mais aussi dans la carte de profondeur.

La Fig. 5.13 illustre le phénomène de façon schématique avec une translation horizontale du point de vue : les parties connues (en bleu et rouge) ne sont que partiellement rendues dans l'image synthétisée. Une partie non négligeable de la vue virtuelle est inconnue (en vert) : l'arrière-plan qui était caché par l'objet de l'avant-plan, et l'arrière-plan qui était en dehors de l'écran.

Comme l'a souligné Yoon *et al.* dans [YOON14], la complétion de cartes de profondeur correspondantes à des vues resynthétisées n'est définitivement pas une tâche facile, et nécessite l'élaboration d'algorithmes dédiés : utiliser une carte de profondeur bien reconstruite permet en effet de guider par la suite le processus de désoccultation d'images proprement dit de manière plus flexible et satisfaisante.

Dans nos travaux [BUYASSE15C], nous avons proposé une analyse poussée du problème de l'inpainting de cartes de profondeur et avons montré que les approches de l'état de l'art ne gère pas les cas géométriques complexes de manière satisfaisante. Nous avons également élaboré un

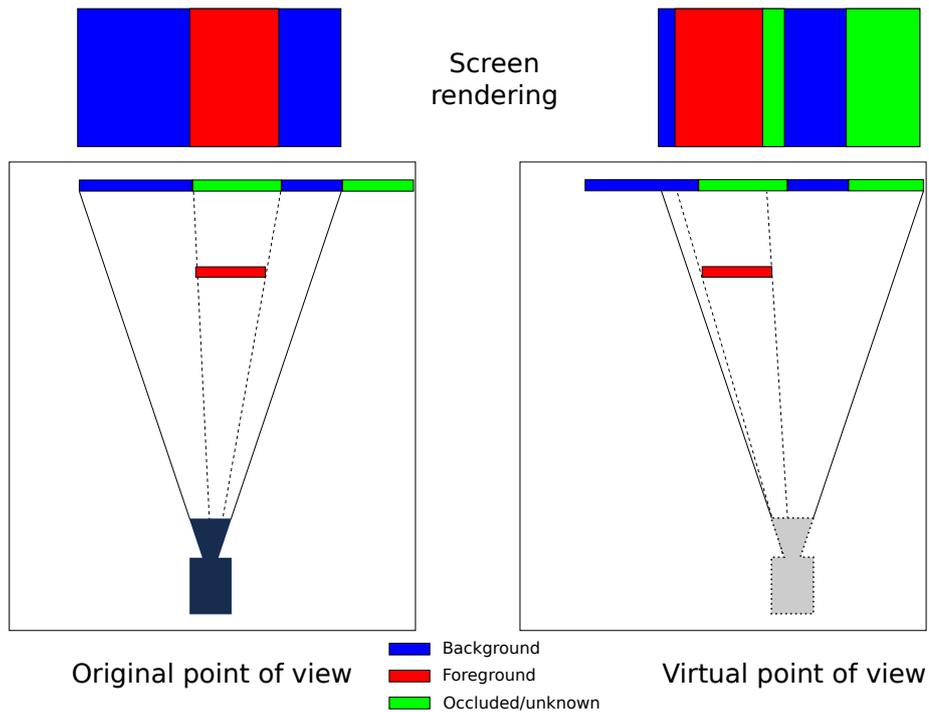


FIG. 5.13: Illustration de la synthèse de vue (image de droite) pour une translation horizontale du point de vue (image de gauche). Les rendus correspondant à l'écran sont donnés en haut de la figure. Le rendu contient notamment beaucoup de zones à restaurer (en vert).

algorithme de complétion de cartes de profondeur efficace. Dans la suite de ce chapitre, nous ne détaillerons pas cet algorithme, et nous supposons donc que ce problème est résolu.

Par contre, nous détaillons les modifications effectuées sur l'algorithme d'inpainting d'images 2D décrit précédemment, pour la désoccultation efficace d'images couleurs correspondants à des points de vues resynthésés, qui prend en compte les données des cartes de profondeur. Le détail de cette méthode d'inpainting fait apparaître des heuristiques astucieuses pour introduire l'information 3D de la carte de profondeur associée dans les méthodes basées motifs, telles que celle décrites dans les sections précédentes. Nous comparons finalement notre approche aux méthodes les plus efficaces de l'état de l'art, avec des résultats de comparaisons qualitatifs et quantitatifs sur des images de références difficiles du domaine.

Bien sûr, nous nous basons sur l'algorithme basé motif de Criminisi *et al.* [CRIMIN04]. La méthode de désoccultation d'images couleurs par patchs sensible à la profondeur, revisite chacune des étapes clés de l'algorithme original, en incorporant l'information de profondeur d'une manière intuitive et efficace. L'approche proposée ne rajoute qu'un seul paramètre λ à ces approches d'inpainting pour les images 2D classiques. Ce paramètre "seuil" λ permet simplement de discriminer les pixels adjacents entre pixel de fond ou pixel d'avant-plan, relativement à leurs profondeurs respectives. Plus particulièrement, deux pixels adjacents p et q sont supposés appartenir au même *objet 3D* (en avant-plan ou arrière-plan) si $|\text{depth}(p) - \text{depth}(q)| < \lambda$.

Nous passons ici en revue les méthodes de la littérature pour l'inpainting d'images RGB-D avec point de vue resynthésés. Nous proposons une classification de ces méthodes en fonction de la modalité traitée : la plupart des approches de l'état de l'art s'occupe du problème de désoccultation d'images couleurs en supposant que les cartes de profondeur correspondantes ont déjà été reconstruites. Dans ces travaux, quelques uns proposent une approche spécifique

en deux étapes (inpainting de la carte de profondeur, suivi de l'inpainting de l'image), tandis que d'autres réalisent les inpainting des deux modalités de façon conjointe. Notons que nous ne passons en revue ici que les méthodes de l'état de l'art qui s'intéressent aux cas de vues virtuelles générées par des warpings horizontaux. Certaines méthodes gèrent des transformations plus génériques (telles que les translations du point de vue en profondeur, suivant z par exemple) [MAO13, MAO14], transformations dont nous ne nous sommes pas occupés.

5.4.1 Revue de l'état de l'art

Deux angles d'attaque ont principalement été étudiés afin d'ajouter l'information de profondeur à l'algorithme initial : (a) modifications du terme de priorité (étape 1 de l'algorithme de Criminisi *et al.* détaillé dans les sections précédentes), et (b) modifications du processus de recherche du meilleur patch (étape 2).

Modifications du terme de priorité. Les auteurs de [DARIBO10] ont proposé d'ajouter un troisième terme multiplicatif au terme de priorité $P(p) = C(p) \times D(p) \times L(p)$ où $L(p)$ est un terme de régularité de profondeur, défini comme l'inverse de la variance de profondeur du patch Ψ_p^d centré en p :

$$L(p) = \frac{|\mathcal{N}_p|}{|\mathcal{N}_p| + \sum_{q \in (\mathcal{N}_p \cap (\mathcal{I} \setminus \Omega))} \left(\Psi_p^d(q) - \overline{\Psi_p^d} \right)^2} \quad (5.19)$$

où Ψ_p^d est le patch défini sur la carte de profondeur J_s centré en p et $\overline{\Psi_p^d}$ est la profondeur moyenne de Ψ_p^d . Ce terme additionnel favorise les pixels p qui sont dans des régions homogènes ($L(p) \simeq 1$) et laisse les pixels se trouvant près de la bordure d'un objet ($L(p) \ll 1$) pour la fin de l'inpainting.

De façon similaire, un terme additionnel au terme de priorité est proposé dans [MA12] :

$$P(p) = C(p) \times D(p) \times \left(1 - \frac{\overline{\Psi_p^d}}{z_{max}} \right) \quad (5.20)$$

où $\overline{\Psi_p^d}$ est la profondeur moyenne des pixels de Ψ_p^d et où z_{max} désigne le maximum global des profondeurs de la carte de profondeur.

Dans [NDJIKI11], le terme de priorité est inchangé, mais l'ensemble des pixels $p \in \delta\Omega$ est discrétisé entre pixels de l'avant-plan et pixels de l'arrière-plan, et seuls ceux appartenant à l'arrière-plan reçoivent une priorité effective (les autres ayant une priorité de 0).

De manière similaire, les auteurs de [GAUTIE11] proposent de mettre à 0 la priorité des pixels se trouvant sur le bord droit du trou si la caméra a bougé de droite à gauche. L'idée derrière cette astuce est que, dans ce cas, les trous apparaissent à gauche des objets du premier-plan, et en mettant à 0 leur priorité, cela évite à l'inpainting de commencer du côté des objets du premier-plan. En plus de ce schéma ne donnant des priorités qu'à un seul côté du trou, les auteurs de [GAUTIE11] proposent également d'utiliser des tenseurs de structure 3D [DI ZEN86] pour le calcul du terme de données en y incluant l'information de profondeur. Ce terme de données repose ainsi à la fois sur des caractéristiques couleur et structurelle.

Les auteurs de [XU13] proposent de modifier l'ordre de remplissage de sorte que l'algorithme commence à partir de l'arrière-plan tout en favorisant la continuation des structures qui entrent dans le masque :

$$P(p) = F(p) \times D(p) \times M(p) \times \frac{|\Psi_p|}{|\mathcal{N}_p|} \quad (5.21)$$

où $F(p)$ est une fonction binaire telle que $F(p) = 0$ si p appartient au premier plan, et $F(p) = 1$ si p appartient à l'arrière-plan, et $M(p)$ est un terme additionnel calculé en p par

$$M(p) = \frac{\sum_{r \in \mathcal{N}_p \cap \Omega} \sum_{q \in (\mathcal{N}_p \cap (\mathcal{I} \setminus \Omega))} e^{-\frac{(J(r) - J(q))^2}{2\sigma}}}{|\Psi_p| (|\mathcal{N}_p| - |\Psi_p|)}$$

Finalement, les auteurs de [YOON14] proposent de calculer le terme de priorité par :

$$P(p) = C(p)^\alpha \times D(p)^\beta \times E(p)^\gamma \quad (5.22)$$

où $C(p)$ et $D(p)$ sont les termes de confiance et de données inchangés, $E(p)$ est l'inverse de la carte de disparité, et $\{\alpha, \beta, \gamma\}$ sont des hyperparamètres de pondération fixés par les auteurs. Au delà de ces hyperparamètres, l'idée derrière le terme additionnel $E(p)$ est de donner une priorité supérieure aux pixels de l'arrière-plan (de faibles valeurs de disparité signifient de grandes valeurs de $E(p)$) par rapport à ceux du premier-plan.

Modifications du processus de recherche du meilleur patch. Au delà des modifications liées au terme de priorité, la plupart des méthodes de la littérature proposent également de modifier la façon dont le meilleur patch est cherché/trouvé (étape 2 de l'algorithme de Criminisi *et al.*).

Les auteurs de [DARIBO10] et [GAUTIE11] incorporent la carte de profondeur au calcul de la SSD en tant que quatrième canal. Tandis que les auteurs de [DARIBO10] pondèrent cet ajout à l'aide d'un paramètre de poids additionnel, les auteurs de [GAUTIE11] considèrent la carte de profondeur comme ayant autant de poids que chacun des 3 canaux de l'image couleur. A noter que ces derniers récupèrent les 5 meilleurs patches et réalisent une combinaison de ces patches pondérés par les SSDs respectives. Ce schéma de recherche du meilleur patch favorise la sélection de patches ayant une profondeur similaire au patch cible Ψ_t sans pour autant éviter la sélection d'un patch du premier-plan

Un autre schéma de recherche consiste à restreindre la recherche aux patches ayant une profondeur inférieure au point à reconstruire. Ce schéma est employé dans [YOON14, NDJKI11] avec un paramètre de tolérance ϵ fixé par les auteurs. En restreignant la recherche du patch candidat aux patches ayant une profondeur moyenne inférieure (ou égale), il devient impossible de procéder à la désocclusion de l'arrière-plan avec des données issues du premier-plan, ce qui est une propriété désirée.

Finalement, les auteurs de [XU13] formulent le processus de mise en correspondance de patch comme une combinaison multiplicative entre deux termes de similarité. Le premier, calculé sur la partie non masquée de Ψ_t , est lié à la fois à la similarité entre couleur et profondeur, tandis que le second, calculé sur la partie masquée de Ψ_t , n'est lié qu'à la similarité entre profondeurs.

Ces deux angles d'attaques principalement étudiés dans la littérature impliquent des modifications dans la définition du *terme de priorité* et/ou dans le schéma de recherche des meilleurs

patches. La plupart de ces modifications ne considèrent que le cas de trous simples (un trou qui est entouré uniquement d'arrière-plan et un unique objet d'avant-plan), et ne sont illustrés que sur un ensemble réduit d'expériences. Au delà du fait que les paramètres additionnels ont été réglés manuellement pour chaque image (ce qui peut s'avérer fastidieux en pratique), la robustesse des *termes de priorité* n'est pas réellement garantie avec des images dont la géométrie des structures et des trous est plus complexe (ce qui arrive fréquemment), qui impliquent par exemple de multiples objets à différentes profondeurs qui se chevauchent. En section 5.4.2, nous proposons également quelques modifications simples des termes de priorité et du schéma de recherche des meilleurs patches, mais aussi de la procédure de copier/coller des patches. Nous proposons également un moyen de gérer spécifiquement le problème de l'*aliasing*, ce qui est original puisque à notre connaissance, aucune méthode de l'état de l'art ne s'occupe explicitement de ce problème.

5.4.2 Extensions proposées pour l'inpainting avec carte de profondeur

Dans cette section, nous détaillons une méthode de désoccultation d'une image couleur I_s , correspondant à une vue virtuelle resynthétisée. Dans la suite, nous considérons que la carte de profondeur J_s associée à cette image est connue et ne contient plus de trous. Nous proposons de revisiter chacune des 3 étapes principales de l'algorithme de Criminisi-*et al.* (c-à-d., le calcul des priorités, la recherche du meilleur patch, et la copie de patches) pour utiliser au mieux l'information de profondeur dont nous disposons maintenant.

Terme de priorité prenant en compte la profondeur. Précédemment, nous avons montré que le terme de priorité, et tout particulièrement le sous-terme de données, est un facteur très sensible pour la reconstruction. Le modifier en ajoutant des termes supplémentaires liés à la profondeur, comme ce qui est proposé par la majorité des travaux de l'état de l'art, tend à diminuer sa robustesse. De plus plusieurs approches de l'état de l'art introduisent des termes de pondération qui sont, en pratique, difficiles à régler. Ici, nous proposons d'éviter l'ajout de termes supplémentaires au terme de priorité initial $P(p) = C(p) \times D(p)$. Comme détaillé ci-dessous, nous n'introduisons pas la notion de profondeur dans le terme de *données* $D(p)$, mais nous introduisons celle-ci dans le terme de *confiance* $C(p)$ de manière naturelle et intuitive.

Premièrement, nous choisissons le terme de *données* précédemment proposé fondé sur l'utilisation de tenseurs de structures

$$D(p) = \|\mathbf{G}_p \vec{n}_p\| \quad (5.23)$$

où \vec{n}_p est le vecteur unitaire orthogonal au contour du masque en p , \mathbf{G}_p est la moyenne pondérée des tenseurs de structure estimés sur la partie non-masquée du patch cible Ψ_p :

$$\mathbf{G}_p = \sum_{q \in (\mathcal{N}_p \cap (\mathcal{I}_s \setminus \Omega))} w_q \overrightarrow{\nabla I_q} \overrightarrow{\nabla I_q}^T$$

et w est une fonction gaussienne 2D normalisée, centrée en p . A noter que ce terme de données ne contient *aucune* information liée à la profondeur.

Deuxièmement, nous faisons une constatation simple : le terme de *confiance* est supposé dénombrer l'information *fiable* (pixels connus) autour d'un pixel p , de sorte que ces mêmes pixels puissent être utilisés pour l'étape de recherche du meilleur patch. A ce stade, il paraît donc contre-intuitif de compter des pixels appartenant à l'avant-plan comme *fiables* pour reconstruire

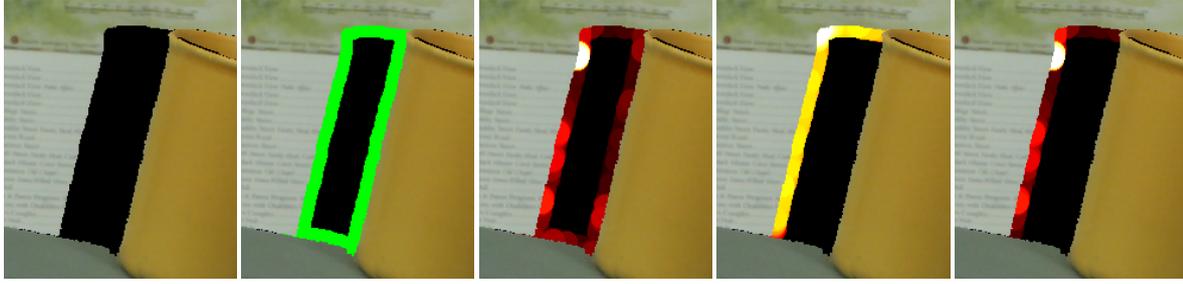


FIG. 5.14: Illustration de l'effet du terme de confiance proposé. De gauche à droite : masque à inpainter, bords intérieurs du masque (en vert), et, en couleurs chaudes, les termes de données, de confiance et termes de priorités finaux. Comme les pixels localisés sur le bord de la tasse ont un terme de confiance nul, leur priorité est également nulle, et le processus d'inpainting commencera par reconstruire l'arrière-plan.

des pixels de l'arrière-plan (et inversement). Nous proposons donc de définir comme *fiabiles* seulement les pixels se trouvant à une profondeur similaire à la profondeur du pixel cible p :

$$C(p) = \frac{1}{|\mathcal{N}_p|} \sum_{\substack{q \in (\mathcal{N}_p \cap (\mathcal{I} \setminus \Omega)) \\ |J_s(p) - J_s(q)| < \lambda}} C(q) \quad (5.24)$$

où $|\mathcal{N}_p|$ est la taille du voisinage \mathcal{N}_p (c-à-d son nombre de pixels) et avec $C(p) = 1, \forall p \in \bar{\Omega}$.

Pour que ce terme de priorité puisse être effectif, une légère modification est envisagée concernant le calcul du contour du masque : au lieu de considérer le contour *extérieur* du masque, nous considérons ici le contour *intérieur*. Cette légère modification fait que, pour un contour du masque se trouvant à la frontière entre l'arrière-plan et l'avant-plan, les pixels du contour se trouveront dans l'arrière-plan. Les pixels voisins appartenant au premier-plan ne seront alors pas pris en compte dans le terme de confiance proposé ci-dessus. La Fig 5.14 montre les avantages de ce terme de confiance et détaille le calcul des termes de priorité pour tous les pixels appartenant au contour d'un trou. Etant donné que les pixels du contour se trouvant près du premier-plan (la tasse) se trouvent dans l'arrière-plan, leur terme de confiance est égal à 0 (Figure 5.14, quatrième image). Ces pixels ont donc un terme de priorité nul, et ne seront considérés par l'algorithme qu'à la toute fin du processus de reconstruction.

Recherche du meilleur patch prenant en compte la profondeur. La stratégie de recherche du meilleur patch est composée de 3 étapes :

- Comme les données de l'image couleur synthétisée I_s sont calculées par *warping* à partir de l'image originale I_o , nous considérons l'hypothèse (raisonnable) que la plupart des données de I_s se retrouvent dans I_o . A partir de là, nous pouvons donc contraindre la recherche du meilleur patch (Eq. 5.1) aux patches de I_o . Grâce à la carte d'offsets (carte de disparité) utilisée lors de la synthèse par *warping*, les fenêtres de recherche optimales peuvent facilement être trouvées dans I_o , en considérant les offsets de *warping* opposés.
- Seuls les patches ayant une profondeur égale à la profondeur de t ($\pm\lambda$) sont considérés comme des patches candidats (étape similaire à celle utilisée dans [NDJIKI11, YOON14]) :

$$\Psi_{\hat{t}} = \left\{ \Psi_p \in I_o \mid \arg \min_{|J_o(p) - J_s(t)| < \lambda} d_{SSD}(\Psi_t, \Psi_p) \right\} \quad (5.25)$$

- Comme dans les sections précédentes, la zone de recherche de patches dans I_o est *divisée* en de nombreuses sous-fenêtres, de façon à utiliser les localisations de patches sources

optimaux trouvés aux itérations précédentes comme point de départ pour la recherche du meilleur patch à copier/coller.

De nombreux avantages découlent de ce schéma de recherche : 1) Grâce à la méthode multi-fenêtrée, la cohérence géométrique globale de la partie reconstruite est meilleure qu'avec un schéma de recherche classique utilisant une fenêtrée unique, 2) Comme les patches sont recherchés uniquement dans I_o , tous les patches candidats trouvés sont complets, 3) Comme les patches candidats ont une profondeur similaire à t , on ne considère effectivement que des patches appartenant le plus souvent au même objet à reconstruire, 4) Tous ces points font que la zone de recherche est fortement réduite, avec un impact très positif sur le temps de calcul effectif de la reconstruction globale.

La Fig.5.15 (droite) montre la zone de recherche (en vert) pour un patch cible appartenant à l'arrière-plan (carré rouge dans l'image de gauche). Les sous-fenêtres proviennent de patches précédemment copiés, et ne contiennent pas les anneaux qui appartiennent au premier-plan. Le patch sélectionné est représenté par le carré bleu (droite).

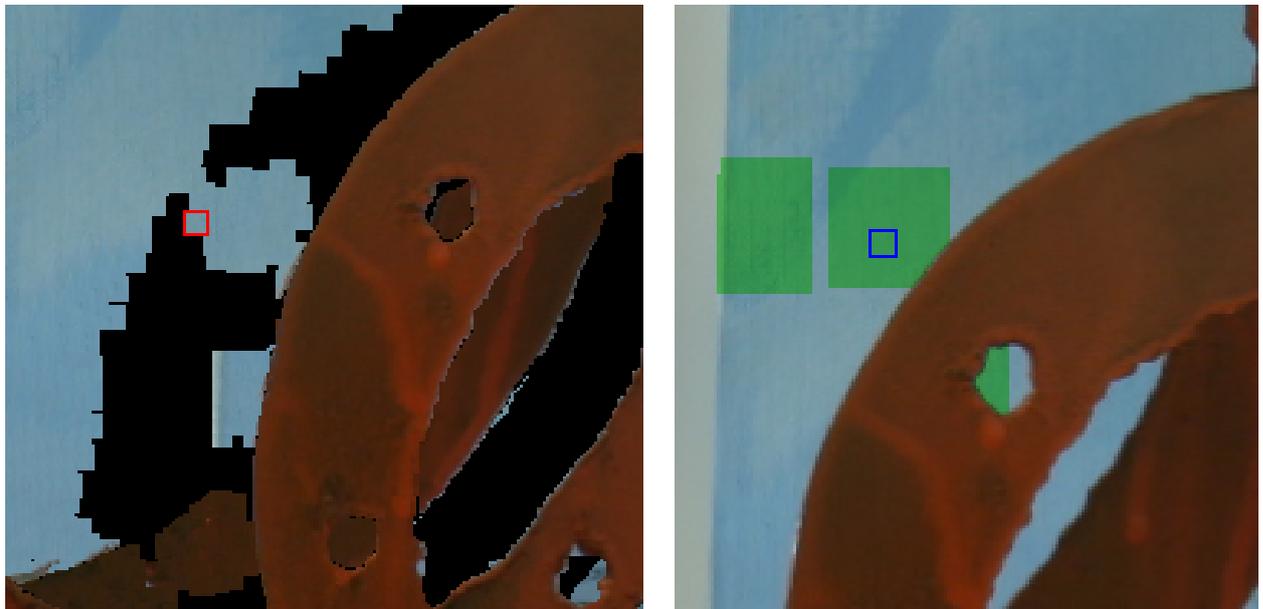


FIG. 5.15: Illustration du schéma de recherche proposé, prenant en compte la profondeur. Gauche : masque à inpainter (en noir), avec le patch cible détourné en rouge. Droite : Espace de recherche (en vert) dans la vue originale, et meilleur patch trouvé (en bleu).

Recopie de patches, prenant en compte la profondeur. Une fois que le meilleur patch a été localisé, les pixels manquants de Ψ_t sont remplis à l'aide des pixels correspondants dans Ψ_t . A notre connaissance, toutes les méthodes de la littérature reposent sur ce schéma simple de recopie. Celui-ci fonctionne bien lorsque le trou à reboucher n'est entouré que d'arrière-plan et d'un seul objet appartenant au premier-plan (le trou ne doit être restauré qu'avec des profondeurs d'arrière-plan). Dans des cas plus complexes où plusieurs objets à différentes profondeurs de chevauchent, il se peut que le trou à combler doive être rempli avec des profondeurs d'arrière-plan *et de premier-plan intermédiaire* (objet se situant entre le fond et l'avant-plan). Dans de tels cas, le schéma classique de copie est inadapté. Afin de mieux gérer ces cas complexes, nous

proposons un schéma de copie prenant en compte la profondeur sous-jacente au masque, qui ne copie les pixels de Ψ_t vers ceux de Ψ_t que si leurs profondeurs respectives sont proches ($\pm\lambda$) :

$$\Psi_t(q) = \Psi_t(p) \left| \begin{array}{l} q - t = p - \hat{t}, \forall q \in \mathcal{N}_t \cap \Omega \\ |J_o(p) - J_s(q)| < \lambda \end{array} \right. \quad (5.26)$$

La Fig.5.16 illustre les bénéfices de ce schéma de copie où le masque à inpainter contient à la fois des données d'arrière-plan mais aussi de l'anneau masqué. Grâce à la stratégie de copie proposée, aucun pixel d'arrière-plan n'est recopié dans une partie correspondant au premier-plan (et vice-versa).

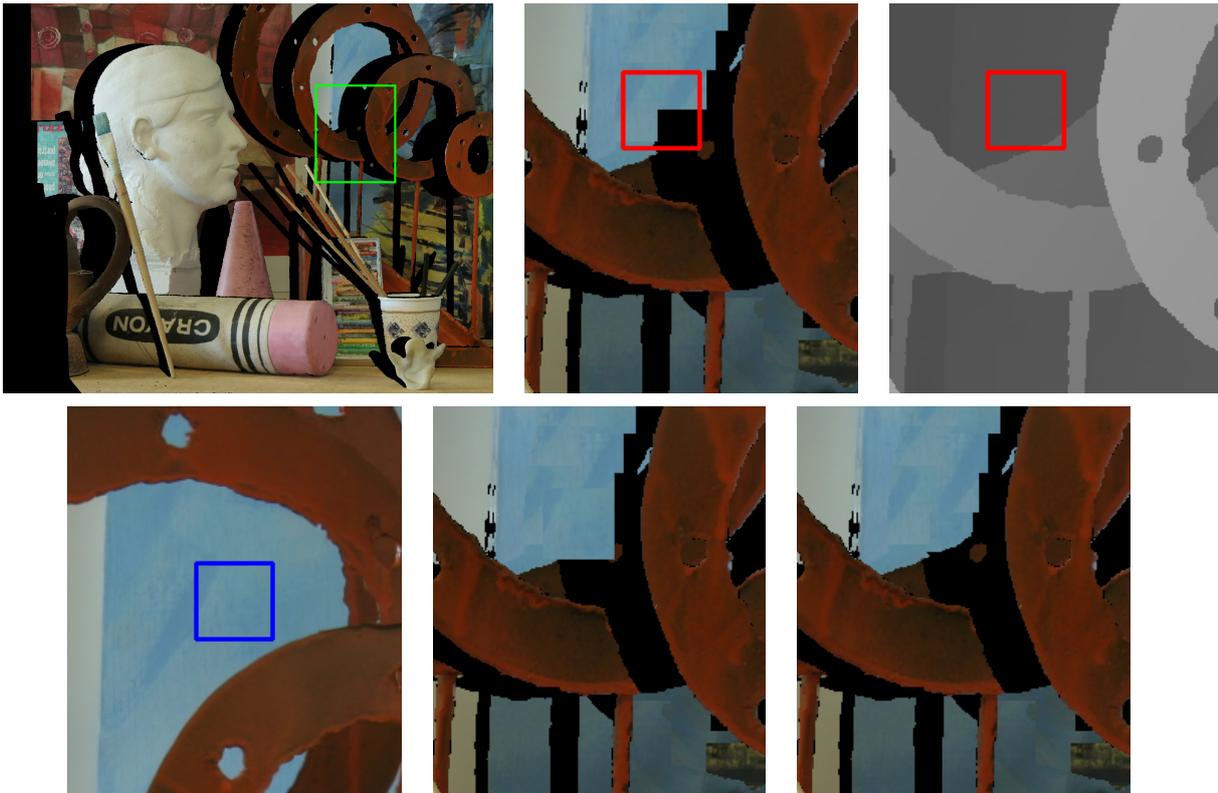


FIG. 5.16: Illustration de la méthode de copie de patch prenant en compte la profondeur pour une itération donnée de l'algorithme d'inpainting. En haut, de gauche à droite : image à reconstruire, zoom de l'image en train d'être inpaintée avec le patch cible en rouge, carte de profondeur correspondante. En bas, de gauche à droite : Meilleur patch trouvé quelque part dans l'image originale (en bleu), résultat d'un schéma de copie de patch classique, et résultat de la méthode de copie de patch que nous proposons. On peut remarquer qu'aucun pixel bleu (du fond) n'est copié à l'intérieur de l'anneau avec notre approche.

5.4.3 Résultats de complétion de vues virtuelles

Nous comparons notre méthode de désoccultation d'images couleurs à deux méthodes représentatives de l'état de l'art [GAUTIE11, YOON14].

Nous utilisons les images de la base de données *Middlebury* [SCHARS14] La Fig. ?? montre des résultats qualitatifs obtenus à l'aide de notre méthode ainsi que des extraits d'images reconstruits obtenus avec les méthodes proposées dans [GAUTIE11] et [YOON14]. Afin de se comparer

de manière équitable, nous utilisons les mêmes cartes de profondeur sous-jacentes pour chaque méthode : la vérité terrain (disponible dans la base de données) pour les images *Art*, *Dolls*, *Midd2* et *Moebius* images (lignes 2, 4, 5, et 6), et la carte de profondeur issue de notre algorithme dédié de désoccultation de carte de profondeur pour les images (voir [BUYASSE15C] pour des détails sur cette partie) pour les images *Adirondack* et *Backpack* (lignes 1 et 3). Les paramètres ont été réglés manuellement pour chaque image pour donner le meilleur rendu visuel possible.

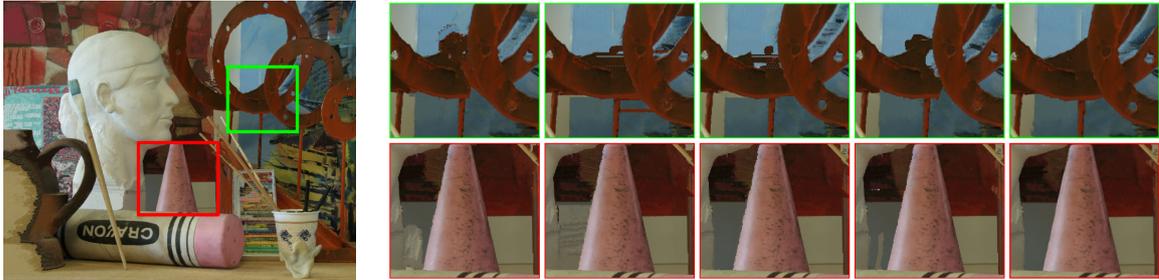


FIG. 5.17: Comparaison de résultats d'inpainting utilisant tous le même algorithme d'inpainting d'images couleurs (le nôtre), mais avec différentes cartes de profondeur reconstruites par différents algorithmes de la littérature. De gauche à droite : résultat global obtenu avec notre méthode, puis extraits d'images inpaintées en utilisant les cartes de profondeurs obtenues avec les méthodes de reconstruction de [DARIBO10], [NDJIKI11], [YOON14], [BUYASSE15C], et notre algorithme.

En pratique, nos résultats d'inpainting prenant en compte les cartes de profondeur génèrent moins d'artéfacts visuels que d'autres méthodes de l'état de l'art. La prise en compte de l'information de profondeur est bien gérée, à tous les niveaux de l'algorithme, ce qui évite beaucoup de configurations de reconstructions géométriquement incohérentes. Il est également important de noter que la méthode de reconstruction proposée ici est en pratique très rapide en temps de calcul, et peut de plus se paralléliser relativement aisément. Comme la recherche des meilleurs patches se réalise dans l'image I_o seulement (qui ne contient aucune donnée manquante), on peut imaginer pouvoir reboucher plusieurs trous distincts en parallèle, plutôt que séquentiellement. De plus, comme l'espace de recherche des meilleurs patches utilisent plusieurs sous-fenêtres de recherche, la recherche d'un meilleur patch $\Psi_{\hat{i}}$ peut également s'envisager de manière parallèle pour accélérer le processus.

La méthode que nous avons proposée ajoute un unique paramètre supplémentaire λ aux méthodes d'inpainting classiques pour prendre en compte l'information de profondeur. Le λ -seuillage, utilisé dans différentes étapes de notre approche dépend essentiellement de la façon dont sont définies les valeurs de la carte de profondeur, et λ est donc un paramètre en pratique simple à régler. Dans toutes les expériences illustrées dans cette section, nous l'avons fixé à la même valeur $\lambda = 4$ puisque les images utilisées proviennent toutes de la même base de données. Choisir un λ trop élevé supprime la différenciation entre les différents plans de profondeur, et on se retrouve dans un cas proche de l'inpainting 2D classique sans prise en compte de la profondeur. A l'inverse, choisir un λ trop faible implique que de toutes petites variations de profondeur seront interprétées comme des contours d'objets 3D. Le rôle de ce paramètre est donc important dans le processus global de reconstruction, et le fait qu'il soit intuitif, et en pratique facile à régler est un avantage indéniable.

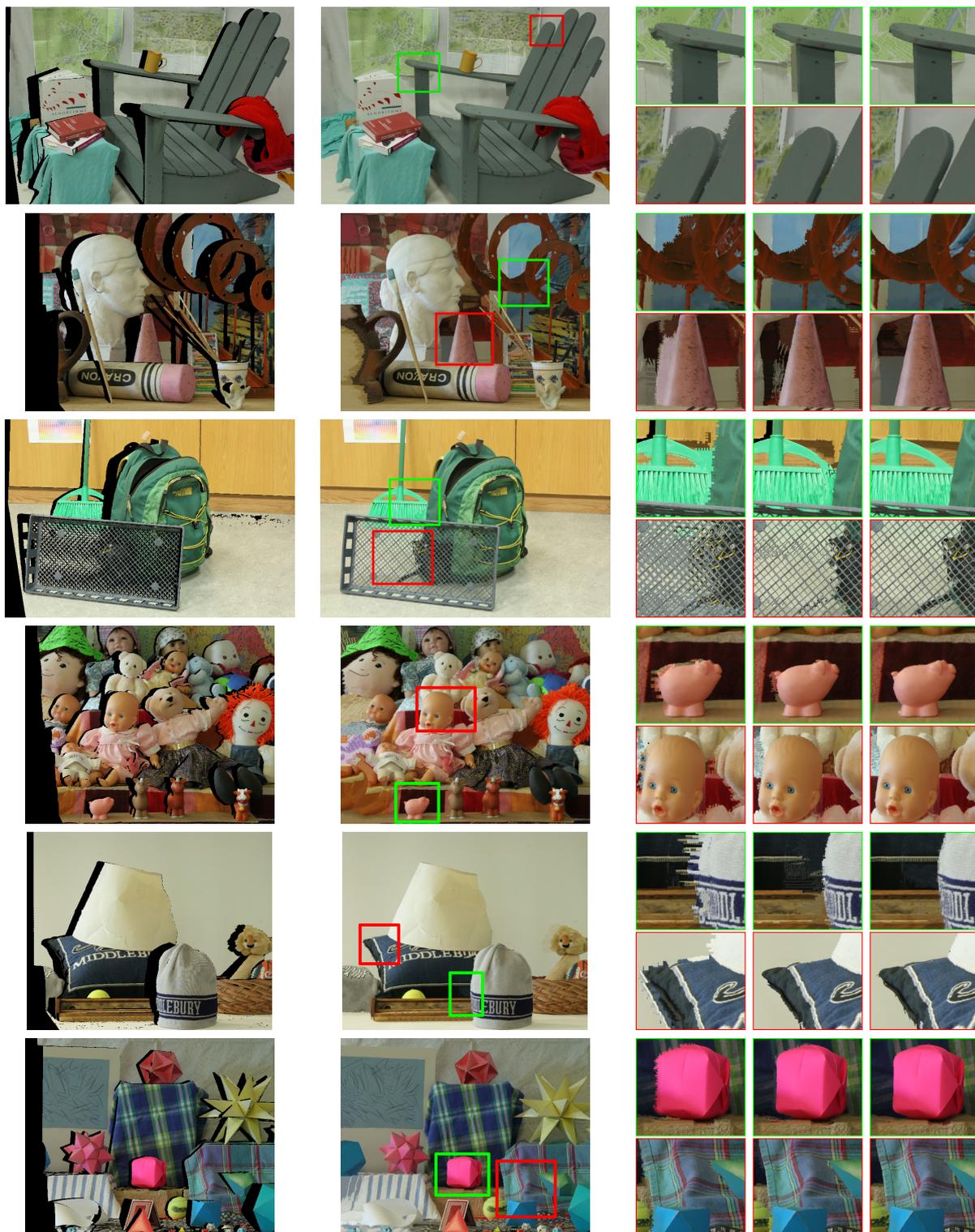


FIG. 5.18: Comparaisons de résultat d'inpainting avec carte de profondeur. De gauche à droite : image masquée, notre résultat, et des extraits de résultats obtenus avec les méthodes [GAUTIE11], [YOON14], et notre méthode. Les tailles des régions à reconstruire sont (de haut en bas), 526K pixels pour l'image *Adirondack*, 401K pixels pour l'image *Art*, 837K pixels pour l'image *Backpack*, 263K pixels pour l'image *Dolls*, 157K pixels pour l'image *Midd2*, et 253K pixels pour l'image *Moebius*.

5.5 Conclusion du chapitre

Nous avons montré dans ce chapitre que l'utilisation de champs tensoriels dans les algorithmes d'inpainting basé motifs apporte un plus indéniable à la reconstruction effectuée pour des images couleurs. En outre, on s'aperçoit que ces modèles tensoriels sont suffisamment génériques pour nous avoir permis d'étendre l'application de l'inpainting guidé tenseurs à des modalités d'images de dimensions supérieures : séquences vidéos et images stéréoscopiques avec prise en charge de cartes de profondeur. Cette propriété des tenseurs à s'adapter naturellement à des problèmes de plus grandes dimensions est d'ailleurs au cœur du chapitre suivant, où nous les utilisons pour la modélisation géométriques de structures dans des espaces de patches, qui sont par nature de très grande dimension.

Régularisation non-locale utilisant des tenseurs en grande dimension

Sommaire

6.1 Combiner EDP et méthodes basées patches ?	87
6.2 Une définition d'un espace de patches continu	89
6.3 Débruitage d'images par régularisation de Tikhonov basée patch	90
6.4 Analyse de la géométrie basée patch, dans Γ	93
6.5 Diffusion anisotrope dans l'espace des patches	94
6.6 Application et résultats	95
6.7 Conclusion du chapitre	96



Nous proposons une famille d'algorithmes de lissage non-local d'images, approximant l'application d'EDP de diffusion dans un espace Euclidien de patches d'images. Nous projetons d'abord l'image à traiter dans cet espace de grande dimension, et nous en estimons une géométrie locale grâce à une extension multi-dimensionnelle des tenseurs de structure. L'analyse des éléments propres de ces tenseurs nous sert ensuite à diriger une EDP de diffusion anisotrope basée patches, qui a à la fois un comportement de régularisation local et non-local, et dont la solution peut s'estimer par des successions de convolutions locales orientées. Nous montrons en particulier que le Filtrage Bilatéral et l'algorithme des Moyennes Non-Locales sont les cas isotropes de notre formalisme général de lissage non-local par patches.

Période : 2009.

Publications associées :

- Chapitres de livre : [BC4].
 - Conférences internationales : [C22] (ICIP'09), [C19] (LNLA'08).
 - Conférences nationales : [NC10] (GRETSI'09).
-

6.1 Combiner EDP et méthodes basées patches ?

Ces dernières années ont vu un regain d'intérêt certain pour les méthodes basées sur l'analyse des patches en traitement d'images. Un patch, défini simplement comme un voisinage local

(souvent carré) d'une image, est un modèle très simple à manipuler mais qui a la capacité intrinsèque de capturer des informations sur les structures non-locales des images (comme les textures par exemple). Les algorithmes basés patches sont relativement intuitifs : souvent, ils reproduisent la façon dont un humain peut appréhender certaines tâches de vision, en analysant et comparant semi-localement des régions d'images entre elles. Ces algorithmes sont généralement assez simples à implémenter et fournissent des résultats souvent remarquables. Ils sont notamment connus pour donner de bonnes solutions aux problèmes de l'estimation de champ de déplacement entre deux images (par *mise en correspondance de bloc*), [HAWORT01], de la synthèse de texture à partir de modèles [ASHIKH01, EFROS99, WEI00], du transfert de textures entre images [ASHIKH03, HERTZM01B] ou encore de l' "inpainting" d'images [CRIMIN04, HAYS07]. De très bons résultats ont été également obtenus récemment pour le *débruitage d'images*, avec l'introduction de la méthode des *Moyennes Non-Locales* [BUADES05A], suivie entre autres par [BROX07, DABOV07, KERVRA06]. Ces méthodes opèrent principalement en calculant des moyennes pondérées de patches à différents endroits de l'image. L'excellente qualité des résultats obtenus a très vite placé ces algorithmes dans le peloton de tête des techniques de lissage/débruitage d'images.

Clairement, ces méthodes concurrencent sérieusement d'autres techniques de régularisation d'images déjà bien établies, comme celles basées sur les EDP de diffusion anisotropes par exemple. Un de leurs grands avantages est leur *non localité* intrinsèque : elles sont capables de prendre en compte des informations images spatialement étendues et les aspects d'auto-similarité de celles-ci. Comme le bruit est un processus stochastique aléatoire, analyser plusieurs échantillons de régions similaires aide énormément à reconstruire des versions dénuées de bruit de ces régions [KERVRA06]. A l'inverse, la régularisation par EDP agit à *une échelle plus locale* en simulant des processus physiques de diffusion des pixels. [PERONA90, SAPIRO01, TSCHUM05, WEICKE98]. Mais cette localité va aussi de pair avec un contrôle très précis du comportement *anisotrope* du lissage. Celui-ci est localement ajustable et permet souvent de prendre en compte des contraintes spécifiques aux applications visées.

Combiner le meilleur de ces mondes locaux et non-locaux est un challenge stimulant, et nous souhaitons ici aller dans ce sens, en proposant une méthode de régularisation basée sur la construction d'un *espace de patches d'images*, de grande dimension, sur lequel l'image dégradée est projetée. Chaque patch existant de l'image d'origine est projetée (continuellement) en un point unique de cet espace, telle que la géométrie de la surface obtenue reflète à la fois les informations structurelles *locales* et/ou *non-locales* des régions de l'image. Cette géométrie est estimée par le calcul d'un champ de tenseurs de structure multi-dimensionnels qui est l'extension naturelle des techniques déjà proposées dans [DI ZEN86, WEICKE98]. Cela nous amène à l'élaboration d'un processus de lissage non-local orienté par ces tenseurs et exprimé sous la forme d'une EDP de diffusion anisotrope multi-valuée dans l'espace des patches de l'image.

Pratiquement, nous montrons que la solution de cette EDP peut s'approximer par des séries de convolutions locales exprimées dans le domaine original de l'image, ce qui évite la construction explicite de l'espace des patches. Nous montrons également que les cas *isotropes* de notre formalisme reviennent aux algorithmes bien connus des *Moyennes Non Locales* et du *Filtrage Bilatéral* [BUADES05A, TOMASI98]. Des comparaisons et résultats d'applications pour le débruitage d'images couleurs concluent ce chapitre.

6.2 Une définition d'un espace de patches continu

Considérons une image 2D bruitée $\mathbf{I} : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^n$ ($n = 3$, pour les images couleurs). La i^{me} composante de l'image multi-valuée \mathbf{I} est notée I_i . Nous définissons un patch d'image $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ comme l'ensemble des valeurs de l'image qui se trouvent dans un voisinage spatial local de \mathbf{I} , de taille $p \times p$ et centré en (x, y) . Pour des raisons de simplicité, la discrétisation spatiale d'un patch sera considérée comme étant réalisée avec un pas unitaire. La taille p est supposée impaire, *c-à-d.* $p = 2q + 1$ ($q \in \mathbb{N}$). En pratique, l'ensemble des valeurs composant un patch $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ peut être ordonné en un vecteur de dimension np^2 :

$$\mathcal{P}_{(x,y)}^{\mathbf{I}} = (I_{1(x-q,y-q)}, \dots, I_{1(x+q,y+q)}, I_{2(x-q,y-q)}, \dots, I_{n(x+q,y+q)})$$

Un patch multi-valué $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ est ainsi vu comme la concaténation de plusieurs patches scalaires $\mathcal{P}_{(x,y)}^{I_i}$ pour toutes les composantes $i = 1 \dots n$ de l'image multi-valuée \mathbf{I} . L'étude mathématique de la géométrie des variétés formées par tous les patches d'une image a été initiée dans [PEYR09]. Ici, nous allons plutôt nous focaliser sur la géométrie "patchiale" de l'image, mais à une échelle plus locale.

Projection de l'image dans l'espace des patches : Nous considérons l'espace euclidien $\Gamma = \Omega \times \mathbb{R}^{np^2}$ de dimension $(np^2 + 2)$ comme un espace de patches localisés. Chaque point \mathbf{p} de Γ est un vecteur de grande dimension dont les coordonnées peuvent stocker des informations à la fois sur des coordonnées (x, y) de Ω et sur toutes les valeurs d'un patch \mathcal{P} quelconque de taille $p \times p$ (donc dans \mathbb{R}^{np^2}). En pratique, nous cherchons à mettre en valeur les points $\mathbf{p} = (x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}})$ de Γ , *c-à-d.* les localisations qui correspondent effectivement à des patches existants dans \mathbf{I} (que nous appelons les *patches localisés* de \mathbf{I}). Nous définissons donc la fonction $\tilde{\mathbf{I}}$ dans Γ telle que $\tilde{\mathbf{I}}_{(\mathbf{p})}$ soit non nulle seulement pour ces patches localisés :

$$\tilde{\mathbf{I}} : \Gamma \rightarrow \mathbb{R}^{np^2+1}, \quad \text{tel que } \forall \mathbf{p} \in \Gamma, \quad \tilde{\mathbf{I}}_{(\mathbf{p})} = \begin{cases} (\mathbf{I}_{(x,y)}, 1) & \text{si } \mathbf{p} = (x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}}) \\ \vec{0} & \text{sinon} \end{cases} \quad (6.1)$$

L'application \mathcal{F} telle que $\tilde{\mathbf{I}} = \mathcal{F}(\mathbf{I})$ calcule donc une sorte de représentation par patch de l'image \mathbf{I} . Notons que l'espace de valeurs de $\tilde{\mathbf{I}}$ a un canal additionnel fixé à 1 pour les patches localisés de \mathbf{I} (à 0 sinon). Cette valeur additionnelle est comparable à ce qui est fait lorsque l'on travaille avec des espaces projectifs : Cette valeur va jouer un rôle de pondération quand on va inverser \mathcal{F} , *c-à-d.* retrouver l'image \mathbf{I} à partir de sa représentation patchiale $\tilde{\mathbf{I}}$. Intuitivement, cette valeur donne une information sur le fait qu'un patch de Γ soit *significatif*, et par défaut, tous les patches localisés de l'image initiale sont considérés de la même importance. Notons également que $\tilde{\mathbf{I}}$ est une fonction multi-valuée non continue. Pour éviter des problèmes de dérivation ultérieurs, nous allons en fait considérer $\tilde{\mathbf{I}}_\epsilon = \mathbf{I} * G_\epsilon$, une version continue de $\tilde{\mathbf{I}}$, où à chaque patch $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ de \mathbf{I} correspond une fonction gaussienne normalisée G_ϵ avec un petit écart type ϵ (proche de 0), au lieu d'un point de Dirac dans Γ .

Rétro-projection dans le domaine image Ω : Du fait de la grande dimensionnalité de l'espace Γ , il n'y a évidemment pas une façon unique d'inverser la transformée en patch $\tilde{\mathbf{I}} = \mathcal{F}(\mathbf{I})$ définie précédemment. Nous définissons donc une méthode de rétro-projection basée sur deux étapes

distinctes : Dans un premier temps, nous recherchons le patch le plus *significatif* $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ dans Γ , et ce, pour chaque localisation (x, y) de Ω .

Celui-ci est simplement le patch possédant le plus grand “poids” en dernière composante, *c-à-d.*

$$\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}} = \operatorname{argmax}_{\mathbf{q} \in \mathbb{R}^{np^2}} \tilde{I}_{np^2+1}(x, y, \mathbf{q}) \quad (6.2)$$

Notons donc que si l’on ne perturbe que très légèrement la représentation par patches $\tilde{\mathbf{I}}$ de \mathbf{I} , les chances de retrouver les patch les plus significatifs $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ aux mêmes points que pour la construction de $\tilde{\mathbf{I}}$ sont très grandes, même si les valeurs de pixels des ces patches les plus significatifs ont été modifiés. C’est l’un des intérêts de cette méthode de rétro-projection : d’un point de vue pratique, nous n’avons pas besoin d’effectuer explicitement une recherche exhaustive des patches les plus significatifs dans l’espace de patch (ce qui serait très fastidieux à réaliser), puisque nous pouvons supposer que les coordonnées de $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ vont se retrouver aux mêmes endroits que les positions initiales des patches (non-filtrés).

Dans un deuxième temps, l’image rétro-projetée $\hat{\mathbf{I}}$ est reconstruite en combinant les patches les plus significatifs ensemble. Ici, nous utilisons la stratégie la plus simple, *c-à-d.* copier le pixel du centre de chaque $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ à sa position (x, y) dans Ω :

$$\forall (x, y) \in \Omega, \hat{I}_{i(x,y)} = \frac{\tilde{I}_{ip^2 + \frac{p^2+1}{2}}(x, y, \mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}})}{\tilde{I}_{np^2+1}(x, y, \mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}})} \quad (6.3)$$

Nous aurions pu également copier un sous-patch entier de $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ en prenant en compte le chevauchement des patches voisins, par rapport à leurs poids respectifs de significativité. Ces considérations de recopie/moyennage de patches apparaissent fréquemment dans les papiers de la littérature des méthodes de traitement d’images basés patch. La meilleure façon de faire dépend en général du type d’applications considéré.

Cette transformée en patch simple nous permet de projeter dans le même temps les informations structurelles locales et non-locales d’une image \mathbf{I} dans un espace Euclidien Γ où les points non-nuls voisins correspondent à des patches de \mathbf{I} qui sont visuellement similaires (dont la distance L_2 , donc la *SSD*, est faible) ou qui sont spatialement proches. Par conséquent, l’idée d’appliquer un processus purement local (par nature) dans Γ , suivi d’une étape de rétro-projection dans Ω semble une façon élégante de manipuler implicitement des propriétés géométries non-locales des images. C’est ce que nous allons chercher à faire dans la suite de ce chapitre, avec l’utilisation de la représentation “patchiale” proposée pour deux applications distinctes (le débruitage d’images, et la mise en correspondance), basées sur l’évolution d’EDP dans Γ .

6.3 Débruitage d’images par régularisation de Tikhonov basée patch

Supposons que l’on dispose d’une image multi-valuée $\mathbf{I}^{\text{bruité}} : \Omega \rightarrow \mathbb{R}^n$ corrompue par un bruit additif quelconque. Ca sera également le cas de sa transformée en patch $\tilde{\mathbf{I}}^{\text{bruité}}$. Nous cherchons ici un flot minimisant basé patch capable de régulariser $\tilde{\mathbf{I}}^{\text{bruité}}$ plutôt que de chercher à régulariser directement $\mathbf{I}^{\text{bruité}}$. Dans cette optique, nous minimisons l’énergie classique E_1

suivante, plus connue sous le nom de fonctionnelle de régularisation de *Tikhonov*, qui a été ici simplement étendue à l'espace de grande dimension Γ :

$$E_1(\tilde{\mathbf{I}}) = \int_{\Gamma} \|\nabla \tilde{\mathbf{I}}_{(\mathbf{p})}\|^2 d\mathbf{p} \quad (6.4)$$

où $\|\nabla \tilde{\mathbf{I}}_{(\mathbf{p})}\| = \sqrt{\sum_{i=1}^{np^2+1} \|\nabla \tilde{I}_{i(\mathbf{p})}\|^2}$ est l'extension habituelle de la norme du gradient pour les données multi-valuées [DI ZEN86]. Notons que ce gradient multi-valué inclut donc aussi le gradient $\|\nabla \tilde{I}_{np^2+1}\|$ des poids des patches.

Flot minimisant : L'EDP qui minimise (6.4) est trouvée par la dérivation de $E_1(\tilde{\mathbf{I}})$ utilisant les équations d'Euler-Lagrange donnant l'expression de l'algorithme de descente de gradient correspondant. Nous retrouvons ici bien entendu l'équation bien connue *de la chaleur*, mais qui s'applique dans notre cas dans l'espace des patches Γ de grande dimension :

$$\begin{cases} \tilde{\mathbf{I}}_{[t=0]} = \tilde{\mathbf{I}}^{\text{bruité}} \\ \frac{\partial \tilde{I}_i}{\partial t} = \Delta \tilde{I}_i \end{cases} \quad (6.5)$$

où Δ est l'opérateur Laplacien dans Γ . De la même manière que pour les techniques de débruitage d'images basées sur les EDP de diffusion classiques [AUBERT02, BARASH02, BARASH04, SAPIRO01, SOCHEN98B, TSCHUM05], nous ne sommes pas particulièrement intéressés par la solution stable de (6.5), puisqu'elle correspond grossièrement à une solution constante. Nous sommes cherchons plutôt à arrêter ce flot de chaleur multi-dimensionnel après un nombre d'itérations *fini* t_1 . Il a été prouvé dans [KOENDE84] que la solution de cette équation est la convolution de l'estimée initiale $\tilde{\mathbf{I}}^{\text{bruité}}$ avec un noyau gaussien normalisé G_σ d'écart type $\sigma = \sqrt{2 t_1}$. Ici, la convolution doit donc être réalisée dans l'espace des patch de grande dimension Γ : $\tilde{\mathbf{I}} = \tilde{\mathbf{I}}^{\text{bruité}} * G_\sigma$, avec

$$\forall \mathbf{p} \in \Gamma, G_{\sigma(\mathbf{p})} = \frac{1}{(2\pi\sigma^2)^{\frac{np^2+2}{2}}} e^{-\frac{\|\mathbf{p}\|^2}{2\sigma^2}} \quad (6.6)$$

Cette expression est donc une façon directe de définir l'équivalent du processus de régularisation de *Tikhonov* pour les patches d'images.

Intérpretation du flot dans le domaine image : Cela devient intéressant quand on essaye d'interpréter le résultat de ce flot de chaleur basé patch (6.5) dans le domaine image original Ω . La transformée en patches (6.1) nous dit que $\tilde{\mathbf{I}}^{\text{bruité}}$ s'annule presque partout dans Γ , excepté sur les points $\mathbf{p} = (x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}^{\text{bruité}}})^T$ correspondant aux positions des patches de l'image originale $\mathbf{I}^{\text{bruité}}$. Donc, le calcul de la convolution (6.6) peut être simplifié en : $\forall (x, y, \mathcal{P}) \in \Gamma$,

$$\tilde{\mathbf{I}}_{(x,y,\mathcal{P})} = \int_{\Omega} \tilde{\mathbf{I}}_{(p,q,\mathcal{P}_{(p,q)}^{\mathbf{I}^{\text{bruité}}})}^{\text{bruité}} G_{\sigma(p-x,q-y,\mathcal{P}_{(p,q)}^{\mathbf{I}^{\text{bruité}}}-\mathcal{P})} dp dq$$

Nous remarquons également que les positions des patches les plus significatifs dans $\tilde{\mathbf{I}}$, comme défini par (6.2), seront les mêmes que celles de $\tilde{\mathbf{I}}^{\text{bruité}}$, puisque l'opérateur de convolution (6.6) par un noyau gaussien normalisé ne modifie par la position des maxima des poids de patches \tilde{I}_{np^2+1} [KOENDE84]. La rétro-projection (6.3) de la transformée en patch $\tilde{\mathbf{I}}$ a donc une expression simple dans Ω , et peut être écrite directement à partir de $\mathbf{I}^{\text{bruité}}$, en considérant que :

$$\forall(x, y), \quad \mathcal{P}_{max(x,y)}^{\mathbf{I}} = \mathcal{P}_{max(x,y)}^{\mathbf{I}^{bruité}} = \mathcal{P}_{(x,y)}^{\mathbf{I}^{bruité}}$$

ainsi que

$$\tilde{I}_{ip^2 + \frac{p^2+1}{2}}^{bruité}(x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}^{bruité}}) = I_{i(x,y)}^{bruité}$$

et

$$\tilde{I}_{np^2+1(x,y,\mathcal{P}_{(x,y)}^{\mathbf{I}^{bruité}})}^{bruité} = 1$$

L'utilisation de ces relations ensemble avec (6.3), et après quelques calculs additionnels, nous arrivons à :

$$\forall(x, y) \in \Omega, \quad \mathbf{I}_{(x,y)} = \frac{\int_{\Omega} \mathbf{I}_{(p,q)}^{bruité} w_{(x,y,p,q)} dp dq}{\int_{\Omega} w_{(x,y,p,q)} dp dq} \quad (6.7)$$

avec $w_{(x,y,p,q)} = w_{(x,y,p,q)}^s w_{(x,y,p,q)}^p$, où $w_{(x,y,p,q)}^s = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-p)^2+(y-q)^2}{2\sigma^2}}$ and

$$w_{(x,y,p,q)}^p = \frac{1}{(2\pi\sigma^2)^{\frac{np^2}{2}}} e^{-\frac{\|\mathcal{P}_{(x,y)}^{\mathbf{I}^{bruité}} - \mathcal{P}_{(p,q)}^{\mathbf{I}^{bruité}}\|^2}{2\sigma^2}} \quad (6.8)$$

Ainsi, chaque valeur de pixel $\mathbf{I}_{(x,y)}$ de l'image régularisée est le résultat d'un moyennage local de tous les pixels $\mathbf{I}_{(p,q)}^{bruité}$, avec une pondération qui dépend à la fois de la distance spatiale entre les points (x, y) et (p, q) (premier terme dans (6.8)), et de la similarité entre les patchs correspondants centrés en (x, y) et (p, q) (second terme dans (6.8)). Bien entendu, cette interprétation du processus de *Tikhonov* basé patch comme un filtrage dans Ω simplifie grandement l'étape d'implémentation puisqu'elle suggère naturellement un moyen de calculer le résultat dans un espace Ω de petite dimension. Notons que les fonctions gaussiennes des termes spatiaux et basés patch de la fonction de pondération $w_{(x,y,p,q)}$ (6.8) ont le même écart type σ . Avoir différents σ peut être très facilement obtenu en pre-multipliant les valeurs de l'image $\mathbf{I}^{bruité}$ par un facteur λ avant le traitement, ce qui revient à avoir deux écarts types différents, tels que $\sigma_{patch} = \sigma_{spatial}/\lambda$.

Lien avec d'autres méthodes de filtrage : Cette régularisation basée patch de *Tikhonov* (6.7) est en réalité très similaire à la technique des moyennes non-locales (*Non Local Means*), comme définit dans [BUADES05A]. Les différences sont de deux types : d'abord, notre formulation est naturellement définie comme la solution d'un *flot minimisant* agissant sur des images multi-valuées, alors que la formulation originale a été imaginée explicitement comme un filtre non-linéaire agissant sur des images scalaires. Ensuite, les poids de moyennage $w_{i,j}$ de l'algorithme des moyennes non locales utilisent seulement un critère de similarité entre patchs. Ici, notre fonction de pondération (6.8) considère également les distances spatiales entre les patch pour le moyennage. Notons aussi que, dans le cas extrême où la taille p des patchs est réduite à 1 (un patch est équivalent à un point), notre méthode de régularisation (6.7) se transforme naturellement en une version multi-valuée d'un autre algorithme de filtrage non-linéaire d'images, à savoir le *filtre bilateral* [BARASH02, BARASH04, PARIS06, TOMASI98]. Ces filtres sont connus pour être anisotropes dans le domaine image Ω . Le bénéfice de minimiser une fonctionnelle de *Tikhonov* dans l'espace des patchs Γ plutôt que dans le domaine de l'image Ω est évidente en termes de qualité de régularisation : On évite le comportement typique de lissage spatial isotrope de la régularisation de *Tikhonov* qui tend à sur-lisser les discontinuités (structures à préserver dans les images), comme les contours, les coins et les textures. Une analyse comparative de ces algorithmes de régularisation est menée en section 6.6.

6.4 Analyse de la géométrie basée patch, dans Γ

Pour aller plus loin, on cherche maintenant à lisser $\tilde{\mathbf{I}}$ d'une manière *anisotrope*, au lieu d'utiliser un flot *isotrope* comme celui de *Tikohnov* effectué précédemment. Cela requiert une analyse préliminaire de la géométrie de $\tilde{\mathbf{I}}$ dans Γ , en recherchant les directions et les contrastes des discontinuités locales de $\tilde{\mathbf{I}}$. Pour des images multi-valuées 2D, cette étape peut se réaliser de manière efficace en calculant les éléments spectraux du champ de tenseur de structure (possiblement lissé) \mathbf{J}_σ , comme cela a été suggéré dans [DI ZEN86, WEICKE98]. Or, comme l'expression de \mathbf{J}_σ est indépendante de la dimension de l'espace considéré, on peut tout à fait l'étendre pour des espaces de grande dimension $\tilde{\mathbf{J}}_\sigma = \sum_{i=1}^n \nabla \tilde{I}_{i\sigma} \nabla \tilde{I}_{i\sigma}^T$ pour l'analyse de notre transformée en patch $\tilde{\mathbf{I}}$, où $\tilde{I}_{i\sigma} = \tilde{I}_i * G_\sigma$ est une version lissée de \tilde{I}_i par un noyau gaussien normalisé G_σ de dimension $(np^2 + 2)$.

Estimation de $\nabla \tilde{\mathbf{I}}_\sigma$: En pratique, la grande dimensionalité $(np^2 + 2)$ de Γ est un obstacle majeur à l'estimation de $\tilde{\mathbf{J}}_\sigma$ puisqu'on ne peut raisonnablement pas stocker toutes les valeurs de $\tilde{\mathbf{I}}$ dans une mémoire d'ordinateur. Heureusement, ces gradients peuvent être estimés directement dans le domaine image original Ω grâce à la propriété de dérivation de la convolution $\partial(\tilde{\mathbf{I}} * \mathcal{K}) = \tilde{\mathbf{I}} * \partial\mathcal{K}$, et au fait que $\tilde{\mathbf{I}}$ s'annule presque partout sauf aux coordonnées $(x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}})$. Ainsi, nous avons : $\forall \mathbf{p} \in \Gamma, \nabla \tilde{I}_{i\sigma}(\mathbf{p}) = \int_{(k,l) \in \Omega} I_i(k,l) \nabla G_{\sigma(\mathbf{p}-\mathbf{q}(k,l))} d_k d_l$, où $\mathbf{q}(k,l) = (k, l, \mathcal{P}_{(k,l)}^{\mathbf{I}})$. Une fois que ces gradients lissés $\nabla \tilde{I}_{i\sigma}$ ont été estimés, le calcul du tenseur de structure $\tilde{\mathbf{J}}_\sigma$ (matrice symétrique $(np^2 + 2) \times (np^2 + 2)$) devient une opération triviale.

Analyse de la géométrie : Comme cela a été démontré dans [DI ZEN86], les éléments propres du tenseur de structure $\tilde{\mathbf{J}}_\sigma$ sont étroitement liés aux variations locales de la fonction multi-valuée étudiée. Dans notre cas, $\mathcal{N}_{(\mathbf{p})} = \sqrt{\text{trace}(\tilde{\mathbf{J}}_{(\mathbf{p})})}$ donne de précieuses informations sur les amplitudes de variations locales, en considérant à la fois les distributions spatiales et basées patches des patches dans un voisinage (*c-à-d.* les discontinuités de la géométrie des patches). Une valeur élevée pour $\mathcal{N}_{(\mathbf{p})}$ signifie qu'un pixel est localisé sur un contour de \mathbf{I} , mais aussi que son patch associé est probablement très peu similaire à d'autres patches de l'image.

Le vecteur propre principal $\tilde{\mathbf{u}}$ de $\tilde{\mathbf{J}}_\sigma$ est une autre information précieuse : il donne l'orientation normale aux isophotes de $\tilde{\mathbf{I}}$, et ici, est lié au vecteur normal à la variété formée par les patches localisés de \mathbf{I} dans Γ . On cherche évidemment à éviter le plus possible de lisser $\tilde{\mathbf{I}}$ le long de la direction $\tilde{\mathbf{u}}$ de manière à préserver les discontinuités significatives des structures de patches de \mathbf{I} . C'est une heuristique usuelle en lissage d'images, qui consiste à favoriser le lissage le long des directions des contours plutôt qu'orthogonalement à ceux-ci. Ici, les "contours" sont des contours de patches dans l'espace de patches Γ . L'écart type σ du lissage gaussien de $\tilde{\mathbf{J}}_\sigma$ est relatif à la régularité de la géométrie spatiale/patchiale estimée de $\tilde{\mathbf{I}}$. Idéalement, ce paramètre dépend du niveau de bruit présent dans \mathbf{I} . Là encore, on peut équilibrer en pratique l'importance de la géométrie patchiale par rapport à la géométrie spatiale, en multipliant les valeurs de l'image originale \mathbf{I} par un facteur $\lambda > 0$ pour donner plus d'importance aux géométries spatiale ($\lambda \rightarrow 0$) ou patchiale ($\lambda \gg 0$) analysées.

6.5 Diffusion anisotrope dans l'espace des patches

Nous possédons maintenant une représentation basée patch $\tilde{\mathbf{I}}$ d'une image \mathbf{I} et des indicateurs géométriques locaux $\mathcal{N}_{(\mathbf{p})}$ et $\tilde{\mathbf{u}}_{(\mathbf{p})}$. Nous proposons donc d'appliquer l'EDP de diffusion générique suivante dans Γ :

$$\forall \mathbf{p} \in \Gamma, \quad \frac{\partial \tilde{\mathbf{I}}_{(\mathbf{p})}}{\partial t} = \text{trace} \left(\tilde{\mathbf{D}}_{(\mathbf{p})} \tilde{\mathbf{H}}_{i(\mathbf{p})} \right) \quad (6.9)$$

C'est une extension aux grandes dimensions de l'équation de diffusion anisotrope que nous avons proposé dans [TSCHUM05] pour la régularisation d'images 2D classiques. $\tilde{\mathbf{H}}_{i(\mathbf{p})}$ dénote la matrice hessienne de \tilde{I}_i en \mathbf{p} . $\tilde{\mathbf{D}} : \Gamma \rightarrow \mathbb{P}_{(np^2+2)}$ est un champ de tenseurs de diffusion anisotropes qui dirigent le processus de régularisation en définissant les orientations (vecteurs propres des tenseurs) et les amplitudes (valeurs propres de tenseurs) de la diffusion en chaque point \mathbf{p} de Γ . La façon dont $\tilde{\mathbf{D}}$ est construit est discuté plus loin dans ce chapitre, mais il dépend bien évidemment de $\mathcal{N}_{(\mathbf{p})}$ et $\tilde{\mathbf{u}}_{(\mathbf{p})}$.

Approximation d'une solution par convolutions locales : Nous avons montré dans [TSCHUM05] que pour un champs de tenseurs constants, l'EDP (6.9) correspond à une équation de la chaleur déformée par un champ tensoriel, dont la solution $\tilde{\mathbf{I}}^{[t]}$ à un temps t peut être trouvée comme la convolution de l'image initiale par un noyau gaussien anisotrope, orienté par le tenseur. Ce résultat est également valable dans notre espace de patches : $\tilde{\mathbf{I}}^{[t]} = \tilde{\mathbf{I}}^{[t=0]} * G_t^{\tilde{\mathbf{D}}}$, où

$$\forall \mathbf{p} \in \Gamma, \quad G_t^{\tilde{\mathbf{D}}} = \frac{1}{(4\pi t)^{\frac{np^2+2}{2}}} e^{-\frac{\mathbf{p}^T \tilde{\mathbf{D}}^{-1} \mathbf{p}}{4t}}$$

Lorsque $\tilde{\mathbf{D}}$ n'est *pas constant* (ce qui correspond au cas usuel), l'EDP (6.9) devient *non-linéaire* et trouver une solution explicite du flot n'est plus possible. Pour éviter une résolution traditionnelle du flot (6.9) en utilisant un schéma de différences finies, nous avons proposé dans [TSCHUM05] un calcul approché itératif basé sur des convolutions locales par différents noyaux gaussiens localement orientés. Dans notre cas, cette approximation peut être transposée dans Γ de la façon suivante :

$$\forall \mathbf{p} \in \Gamma, \quad \tilde{\mathbf{I}}_{(\mathbf{p})}^{[dt]} \approx \int_{\mathbf{q} \in \Gamma} \tilde{\mathbf{I}}_{(\mathbf{q})}^{[t=0]} G_{dt(\mathbf{p}-\mathbf{q})}^{\tilde{\mathbf{D}}_{(\mathbf{p})}} d\mathbf{q} \quad (6.10)$$

où dt est le pas de temps discrétisé utilisé pour l'évolution de l'EDP. Notons que comme (6.10) est un processus de moyennage local, l'approximation proposée reste stable même pour des dt larges (ce qui n'est pas le cas avec des schémas d'Euler aux différences finies). Cette approximation est aussi meilleure lorsque le calcul du champ $\tilde{\mathbf{D}}$ ne dépend pas du temps d'évolution t (*c-à-d.* est estimé à partir de $\tilde{\mathbf{I}}^{[t=0]}$) et ne varie pas localement de manière trop importante (*c-à-d.* est relativement régulier). Finalement, il est intéressant de noter que (6.10) ne va pas non plus modifier de façon significative les positions des patches les plus significatifs de $\tilde{\mathbf{I}}$. En conséquence, la procédure de rétro-projection (6.3) est simplifiée puisque $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ et $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ restent à la même position. En prenant tout ceci en compte, on peut approximer la solution de (6.9) en utilisant une itération unique (avec un pas de temps important), et seulement en calculant des convolutions localisées aux coordonnées des patches originaux $(x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}})$, et comme $\tilde{\mathbf{I}}^{[t=0]}$ est épars, on peut le formuler directement dans le domaine image original $\Omega : \forall (x, y) \in \Omega$,

$$\tilde{\mathbf{I}}_{(\mathbf{p}(x,y))}^{[t]} \approx \int_{(k,l) \in \Omega} \mathbf{I}_{(k,l)}^{[t=0]} G_{dt(\mathbf{p}(x,y)-\mathbf{q}(k,l))}^{\tilde{\mathbf{D}}_{(\mathbf{p}(x,y))}} d_k d_l \quad (6.11)$$

où $\mathbf{p}_{(x,y)} = (x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}})$ et $\mathbf{q}_{(k,l)} = (k, l, \mathcal{P}_{(k,l)}^{\mathbf{I}})$.

Choix des tenseurs de diffusion : Remarquons que si $\tilde{\mathbf{D}}$ est choisi comme étant la matrice identité, nous retrouvons le cas précédent de l'équation de la chaleur en grande dimension dans Γ , avec le cas isotrope de la section (6.3). Des tenseurs de diffusion isotropes ne vont pas prendre en compte la géométrie locale de la représentation par patch $\tilde{\mathbf{I}}$ de l'image \mathbf{I} . La diffusion va être réalisée avec la même amplitude dans toutes les orientations de l'espace des patches, incluant $\tilde{\mathbf{u}}_{(\mathbf{p})}$, le vecteur propre principal de $\tilde{\mathbf{J}}_{\sigma(\mathbf{p})}$ (ce qu'on veut à tout prix éviter). Afin de lisser l'image tout en préservant sa géométrie spatiale et patchiale $\tilde{\mathbf{I}}$, nous proposons de considérer le champ suivant de tenseurs de diffusion anisotrope :

$$\forall \mathbf{p} \in \Gamma, \quad \tilde{\mathbf{D}}_{(\mathbf{p})} = \frac{1}{\mathcal{N}_{(\mathbf{p})}} \left(\mathbf{I}_d - \tilde{\mathbf{u}}_{(\mathbf{p})} \tilde{\mathbf{u}}_{(\mathbf{p})}^T \right) \quad (6.12)$$

Ce choix est inspiré par le formalisme classique de la *variation totale* pour les images scalaires [RUDIN92], correspondant à la diffusion anisotrope $\frac{\partial I}{\partial t} = \operatorname{div}(\nabla I / \|\nabla I\|)$ qui peut aussi être exprimée sous la forme $\frac{\partial I}{\partial t} = \operatorname{trace}(\mathbf{D}\mathbf{H})$ avec $\mathbf{D} = \frac{1}{\|\nabla I\|} (\mathbf{I}_d - \eta\eta^T)$ et $\eta = \nabla I / \|\nabla I\|$. Intuitivement, ce choix pour les tenseurs de diffusion (6.12) réduit l'amplitude de la diffusion basée patch aux endroits de fort gradients, et lisse les structures patchiales toujours le long de l'hyperplan localement tangent aux discontinuités de $\tilde{\mathbf{I}}$. En conséquence, la diffusion anisotrope (6.9) modélise une procédure *non-locale préservant la géométrie des patches*. Là encore, pour des patches 1×1 ($q = 0$), notre méthode définit une variante anisotrope du *filtre bilatéral*.

6.6 Application et résultats

Nous avons appliqué les différentes techniques proposées dans ce chapitre pour le débruitage de différentes images couleurs, considérées dans leur espace couleur (R, G, B) original.

Le flot de régularisation anisotrope (6.9) dans l'espace des patches Γ peut être utilisé pour améliorer l'apparence d'images couleurs dégradées (ou d'autres données multi-valuées). La Fig.6.4 compare cette méthode avec quelques algorithmes de régularisation les plus proches, comme le *filtre bilatéral*, les *moyennes non-locales* ou encore la régularisation classique de *Tikhonov*, appliqué sur le domaine image Ω . Du bruit blanc Gaussien synthétique ($\sigma_{\text{noise}} = 20$) a été ajouté à l'image couleur originale *Barbara*. Pour être le plus juste possible, nous n'avons pas appliqué les versions scalaires de ces filtres (ainsi que définit dans les articles originaux [BUADES05A, TOMASI98]), mais leur extension multi-valuée. Cela donne effectivement des résultats de débruitage meilleurs qu'en appliquant ces filtres de façon séparée canal par canal. Le *PSNR* entre l'image non-bruitée et restaurée, ainsi que les paramètres utilisés pour les expériences sont affichés. Pour chaque méthode, les paramètres ont été manuellement choisis pour optimiser le *PSNR* obtenu. Comme le flot proposé (6.9) est assez proche dans son expression finale aux expressions des *moyennes non-locales* et du *filtre bilatéral*, les résultats de débruitage sont au final assez proches en terme de qualité visuelle. Notre méthode se comporte néanmoins un peu mieux sur certaines régions texturées. Cela est dû au fait que notre fonction de pondération (6.8) considère à la fois la distance spatiale et la similarité entre patches. L'amélioration visuelle est néanmoins très subtile, et en pratique, ne compense pas la forte complexité algorithmique ajoutée : En effet, le seul calcul des tenseurs de structure N -D prend déjà environ 20 fois

plus de temps que le calcul d'une simple itération de l'algorithme classique des *moyennes non locales*.

Résultat	PSNR	Paramètres	Référence
(b)	25.33 dB	$\sigma = 0.6$	[BOUHAM07, TIKHON63]
(c)	26.97 dB	$\sigma_s = 10, \sigma_r = 25$	[BARASH02, PARIS06, TOMASI98]
(d)	26.98 dB	$\sigma = 21, \text{patch } 7 \times 7$	[BUADES05A]
(e)	27.03 dB	$\sigma = 21, \text{patch } 7 \times 7$	Ce chapitre

FIG. 6.1: PSNR(Original/Débruité) et paramètres associés aux différentes méthodes, utilisés pour les résultats de la Fig.6.4.



(a) Image originale (avec bruit de motif)



(b) Après utilisation d'une EDP de diffusion anisotrope dans Γ

FIG. 6.2: Régularisation anisotrope (6.7) dans l'espace des patches Γ .



(a) Image originale dégradée



(b) Utilisation d'EDP de diffusion anisotrope dans Ω [TSCHUM05]



(c) Utilisation d'EDP de diffusion anisotrope dans l'espace de patches Γ

FIG. 6.3: Comparaison entre résultats de régularisations anisotropes réalisées dans le domaine spatial Ω , et dans le domaine patchial Γ .

6.7 Conclusion du chapitre

Dans ce chapitre, nous nous sommes essayés à la modélisation d'un espace de patches, et la représentation d'une image couleur dans cette espace. Nous espérons que l'analyse géométrique

de cette représentation patchiale pourrait aider, d'une part à définir un meilleur algorithme de lissage non-local d'images, et d'autre part à fusionner les techniques d'EDP de diffusion anisotrope et les méthode de régularisation non-locales basées patches dans un formalisme commun. Les résultats applicatifs obtenus sont en pratique peu intéressants : le gain de performance de la régularisation obtenue par rapport à la technique classique des *moyennes non-locales* n'est pas suffisante pour faire oublier l'augmentation significative de la complexité algorithmique associée à ce nouvel algorithme. L'interprétation des méthodes de régularisation non-locales comme des flots EDP dans un espace de patches de grande dimension reste cependant séduisante. La vie d'un chercheur n'est malheureusement pas couronnée de succès continuels ! Le chapitre suivant illustre d'ailleurs également cet aspect de notre travail, puisque nous nous sommes intéressés à la modélisation de phénomènes de diffusion complexes des molécules d'eau dans les tissus en modalité d'images DT-MRI. Cette modalité particulière nous a permis en particulier de montrer les limites du modèle tensoriel du second ordre, et de les dépasser grâce à une modélisation géométrique plus fine des phénomènes de diffusion.



FIG. 6.4: Comparaisons de résultats de régularisation d'images avec la méthode proposée (6.7).

Estimation de caractéristiques avancées en IRM de diffusion

Sommaire

7.1 Imagerie par résonance magnétique de diffusion	100
7.2 Reconstruction continue du signal de diffusion	102
7.3 Formalisme pour l'estimation de caractéristiques de la PDF	103
7.4 Application et résultats	105
7.5 Conclusion du chapitre	107



Dans ce chapitre, nous proposons une méthode permettant d'estimer des caractéristiques variées du tissu cérébral à partir de l'acquisition d'un nombre réduit d'échantillons de signaux IRM de diffusion in-vivo. Ces caractéristiques doivent permettre l'étude de la structure locale du tissu cérébral, notamment dans la substance blanche. L'approche proposée est flexible quant à la caractéristique calculée et au nombre d'échantillons disponibles. Elle définit un formalisme générique qui d'une part, unifie de nombreux travaux précédents liés à l'estimation des fonctions de densité de probabilité (PDF) de déplacement en IRM de diffusion, dont l'Imagerie du Tenseur de Diffusion (DTI) et le Q-Ball Imaging (QBI). D'autre part, elle permet aussi de définir et d'estimer de nouvelles caractéristiques originales : "vraie" ODF, probabilité de non diffusion, taille moyenne des cellules, etc. Nous validons notre approche par des comparaisons de résultats avec des méthodes de la littérature, sur des données synthétiques ainsi que des données d'un cerveau humain acquises in-vivo dans un intervalle de temps modéré.

Période : 2007–2010.

Encadrements : Haz-Edine Assemlal (doctorant, 2007-2010).

Financements : Bourse de thèse ministère.

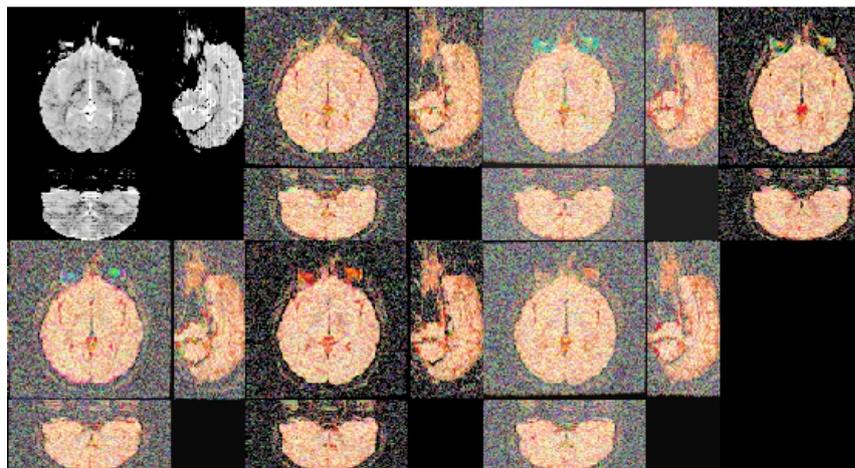
Publications associées :

- Journaux : [J9] (Medical Image Analysis'11), [J8] (Medical Image Analysis'09).
- Conférences internationales : [C23] (MICCAI'09), [C21], [C20] (MICCAI'08), [C18] (ICIP'07).
- Conférences nationales : [NC12] (RFIA'10), [NC11] (GRETSI'09), [NC9] (GRETSI'07).

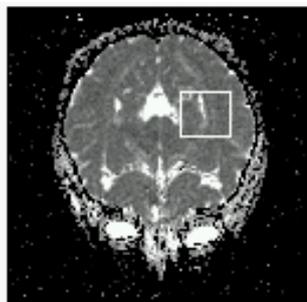
7.1 Imagerie par résonance magnétique de diffusion

L'apparition de l'imagerie IRMd du tenseur de diffusion a soulevé un grand intérêt dans la communauté relative à l'imagerie médicale [BASSER94, BASSER00, LEBIHA86]. Cette modalité d'image 3D non invasive consiste à mesurer le mouvement des molécules d'eau dans les tissus, en utilisant des techniques de résonance magnétique. Elle est basée principalement sur l'acquisition de plusieurs images IRM *brutes* $S_k : \Omega \subset \mathbb{R}^3 \rightarrow \mathbb{R}$ avec différentes séquences de pulses de gradient (orientés dans au moins 6 directions différentes). Une image S_0 est également mesurée, correspondant à une image sans direction de gradient privilégié (Fig.7.1a). Remarquez que ces S_k peuvent être très bruitées. Cela est dû au fait que le temps d'acquisition des IRMs doit être très faible, cette contrainte entraînant fréquemment des artefacts de mesure. Cet ensemble $\{S_k, k = 0 \dots n\}$ de données brutes doit être ensuite estimé en un volume $\mathbf{T} : \Omega \subset \mathbb{R}^3 \rightarrow \mathbb{P}(3)$ de Tenseurs de Diffusion (c-à-d de matrices 3x3 symétriques et définie-positives) qui décrivent à travers leur éléments spectraux, les diffusivités principales $\lambda_1, \lambda_2, \lambda_3$ (avec $\lambda_1 \geq \lambda_2 \geq \lambda_3$) et les directions correspondantes $\mathbf{u}, \mathbf{v}, \mathbf{w}$ (orthogonales) du processus de diffusion des molécules d'eau dans les tissus comme les os, les muscles ou la matière blanche du cerveau (Fig.7.1b).

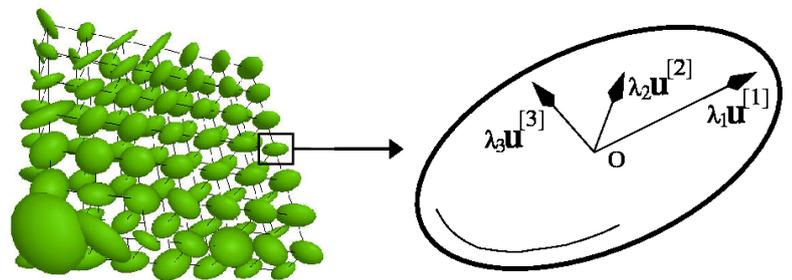
$$\forall x, y, z \in \Omega, \quad \mathbf{T}(x, y, z) = \lambda_1 \mathbf{u}\mathbf{u}^T + \lambda_2 \mathbf{v}\mathbf{v}^T + \lambda_3 \mathbf{w}\mathbf{w}^T$$



(a) Acquisitions brutes mesurées S_k ($k = 0 \dots 6$)



(b1) Diffusivité moyenne $\frac{1}{3}(\lambda_1 + \lambda_2 + \lambda_3)$



(b2) Partie du champ de tenseurs, représentés par des ellipsoïdes 3D.

FIG. 7.1: Principe de l'imagerie IRMd

Selon les caractéristiques du tissu, la diffusion (et donc les tenseurs estimés) peut être isotrope, par exemple dans les régions du liquide céphalo-rachidien, ou anisotrope comme dans la

matière blanche du cerveau où la diffusion est principalement effectuée dans les directions des fibres reliant les neurones entre-eux. L'imagerie IRMd est donc particulièrement adaptée pour étudier la connectivité des neurones à l'intérieur de la matière blanche, en suivant en chaque voxel la direction principale des fibres, donnée par les vecteurs propres principaux $\mathbf{u}(x, y, z)$ des tenseurs $\mathbf{T}(x, y, z)$. Récupérer ces réseaux de fibres implique en pratique de nombreux processus sous-jacents. D'abord une partie d'*estimation*, qui estime les tenseurs de diffusions (matrices 3×3) à partir des données brutes comme étant des modèles Gaussiens de la diffusion des molécules d'eau. Ce champ de tenseurs estimé peut être bruité, et un processus de *régularisation de champs de tenseurs* est alors nécessaire pour améliorer le résultat de l'estimation. Finalement, les fibres doivent être calculées et *visualisées* de manière pratique, ce qui n'est pas trivial étant donné qu'elles sont la plupart du temps condensées dans une même portion d'espace.

Néanmoins, ce modèle tensoriel présuppose que la diffusion locale suit une loi Gaussienne, et ne permet donc pas d'estimer plusieurs orientations de maxima de la diffusion. En réalité, ce cas se produit pourtant fréquemment, par exemple dans des zones de croisements de fibres. Cette limitation du modèle tensoriel DTI implique généralement que le processus de tractographie des fibres de la matière blanche s'arrête lorsque l'on se trouve dans ces zones, par manque d'informations pertinentes sur les directions à suivre (le tenseur de diffusion est alors "rond", sans orientation prédominante). Plusieurs travaux de la littérature ont donc cherché à estimer des modèles géométriques plus sophistiqués pour représenter la grande diversité des comportements diffusifs des molécules d'eau dans les tissus. En théorie, le signal de diffusion E et sa fonction de densité de probabilité (*PDF*) P sont reliés par la transformée de Fourier [CALLAG91A] :

$$P(\mathbf{p}) = \int_{\mathbf{q} \in \mathbb{R}^3} E(\mathbf{q}) \exp(-2\pi i \mathbf{q}^T \mathbf{p}) d\mathbf{q} \quad (7.1)$$

Cependant, des applications cliniques utilisant directement cette relation ne sont pas envisageables [WEDEEN00]. En effet, une estimation de la *PDF* par une transformée de Fourier nécessiterait un échantillonnage complet de l'espace de diffusion (*q-space*), induisant un temps d'acquisition extrêmement long. Afin de contourner ce problème, deux groupes de protocoles d'échantillonnage se sont développés récemment dans la littérature. Ces méthodes se consacrent non plus à l'estimation de l'ensemble de la *PDF*, mais à des caractéristiques de celle-ci, soit radiales, soit angulaires. L'Imagerie de Diffusion à Haute Résolution Angulaire (*HARDI*) s'attache à l'analyse angulaire du signal de diffusion. Dans cette modalité d'acquisition le signal est mesuré sur plusieurs points d'une sphère dans l'espace de diffusion. Plusieurs méthodes basées sur ce protocole ont proposé de calculer une mesure angulaire de la diffusion [DESCOT06, ÖZAR06, TUCH99] à l'aide d'indices comme la fonction de densité d'orientation (*ODF*), définie comme la projection radiale de la *PDF*. Néanmoins l'échantillonnage du signal sur une seule sphère entraîne la perte d'importantes informations sur la microstructure du tissu observé. En effet les modèles estimés sont uniquement des approximations de la "véritable" *ODF*. De plus aucune reconstruction de la diffusion radiale n'est effectuée : cela délaisse un aspect important de l'IRM de diffusion qui a pourtant des applications reconnues [ASSAF98]. Afin de résoudre ce problème, nous proposons une méthode flexible et générique pour le calcul de caractéristiques de la *PDF*. Cette approche est basée sur une estimation sans modèle a priori et indépendante du nombre d'échantillons. On se propose d'estimer ces caractéristiques en deux étapes : la reconstruction de l'image de diffusion sous une forme continue ; suivie d'une projection dans la base choisie entre le signal reconstruit et une caractéristique de la *PDF*.

7.2 Reconstruction continue du signal de diffusion

On définit l'espace de diffusion q -space comme un sous domaine de \mathbb{R}^3 , avec le vecteur de diffusion \mathbf{q} . Le signal de diffusion E est reconstruit de manière continue dans une base sphérique spécifique $\Psi_{n,l,m}$,

$$\hat{E}(\mathbf{q}) = \sum_{n=0}^{\infty} \sum_{l=0}^{\infty} \sum_{m=-l}^l a_{n,l,m} \Psi_{nlm}(\mathbf{q}) \quad (7.2)$$

$$\Psi_{nlm}(\mathbf{q}) = R_n(\|\mathbf{q}\|) y_l^m \left(\frac{\mathbf{q}}{\|\mathbf{q}\|} \right), \quad (7.3)$$

où les a_{nlm} sont les coefficients du signal continu dans la base Ψ . Les y_l^m sont des harmoniques sphériques réelles, R_n sont des fonctions de Gauss-Laguerre.

La partie angulaire du signal de diffusion est capturée par des fonctions définies sur la sphère unité. Les harmoniques sphériques complexes sont solutions de la partie angulaire de l'équation de diffusion de Laplace en coordonnées sphériques. Pour cette raison, ces fonctions ont été largement utilisées en IRM de diffusion. En effet, le sous ensemble des harmoniques sphériques réelles et symétriques est notamment bien adapté à la physique de la diffusion :

$$y_l^m = \begin{cases} \sqrt{2} \operatorname{Re}(Y_l^m) & si \quad 0 < m \leq l \\ Y_l^0 & si \quad m = 0 \\ \sqrt{2} \operatorname{Im}(Y_l^{|m|}) & si \quad -l \leq m < 0 \end{cases} \quad \text{avec } l \in 2\mathbb{Z} \quad (7.4)$$

La partie radiale du signal est reconstruite par les fonctions R_n . Puisque le nombre d'acquisitions radiales est très limité, nous choisissons R_n afin de représenter le signal de façon parcimonieuse, *c-à-d.* avec le minimum de coefficients. Nous proposons une estimation du signal radial sans modèle en se basant sur les fonctions de Gauss-Laguerre :

$$R_n(\|\mathbf{q}\|) = K \exp\left(-\frac{\|\mathbf{q}\|^2}{2\gamma}\right) L_n^{1/2}\left(\frac{\|\mathbf{q}\|^2}{\gamma}\right) \quad (7.5)$$

où $K = [(2n!)/(\gamma^{3/2}\Gamma(n+3/2))]^{1/2}$ est une constante de normalisation, γ représente un facteur d'échelle, $L_n^{(\alpha)}$ sont les polynômes généralisés de Laguerre. La constante d'échelle γ peut être aisément calculée à partir des échantillons du signal. Ainsi, une troncature de R_n à un ordre faible fait l'hypothèse d'une décroissance radiale Gaussienne, alors qu'un ordre de troncature élevé permet l'estimation du signal sans modèle. L'ordre de troncature est relié aux nombres d'échantillons du signal de diffusion. La Fig.7.2 illustre l'adéquation entre la base radiale R_n choisie et les données expérimentales de diffusion mesurées [G. REG10]

En pratique, les ordres de troncature radial N et angulaire L sont choisis pour que le nombre de coefficients dans la base Ψ , $n_c = (N+1)(L+1)(L+2)/2$, soit environ la moitié du nombre d'échantillons n_s , *c-à-d.* , $n_c \leq 2n_s$. Le choix de favoriser N ou L dépend largement de la distribution des échantillons dans le q -space, et donc au final de la caractéristique que l'on souhaite observer : L élevé pour des caractéristiques angulaires (carte d'anisotropie, tractographie), ou N élevé pour des caractéristiques radiales (taille des cellules, absence de myéline).

Nous proposons de reconstruire le signal de diffusion dans la base orthonormée Ψ par la méthode des moindres carrés pondérés. Les coefficients correspondants a_{nlm} sont donnés par la pseudo-inverse de Moore-Penrose régularisée :

$$\mathbf{A} = \arg \min_A \|\mathbf{E} - \mathbf{MA}\|^2 + \lambda_l \|\mathbf{L}\|^2 + \lambda_n \|\mathbf{N}\|^2 \quad (7.6)$$

avec $\mathbf{M} = (R_n(\|\mathbf{q}_j\|)y_l^m(\frac{\mathbf{q}_j}{\|\mathbf{q}_j\|}))_{nlm \times j \in \mathbb{N}^3 \times \mathbb{N}}$, la matrice de base et \mathbf{E} , \mathbf{A} respectivement les vecteurs $(E(\mathbf{q}_1), \dots, E(\mathbf{q}_{n_s}))^T$ et $(a_{000}, \dots, a_{NLL})^T$. Comme la matrice \mathbf{M} est susceptible d'être mal conditionnée dû à un nombre très restreint d'échantillons, nous introduisons les matrices de régularisation \mathbf{L} et \mathbf{N} avec des entrées $l(l+1)$ et $n(n+1)$ le long de leur diagonale. La régularisation angulaire \mathbf{L} pénalise une forme sphérique trop anisotrope. La régularisation radiale \mathbf{N} favorise une décroissance Gaussienne du signal.

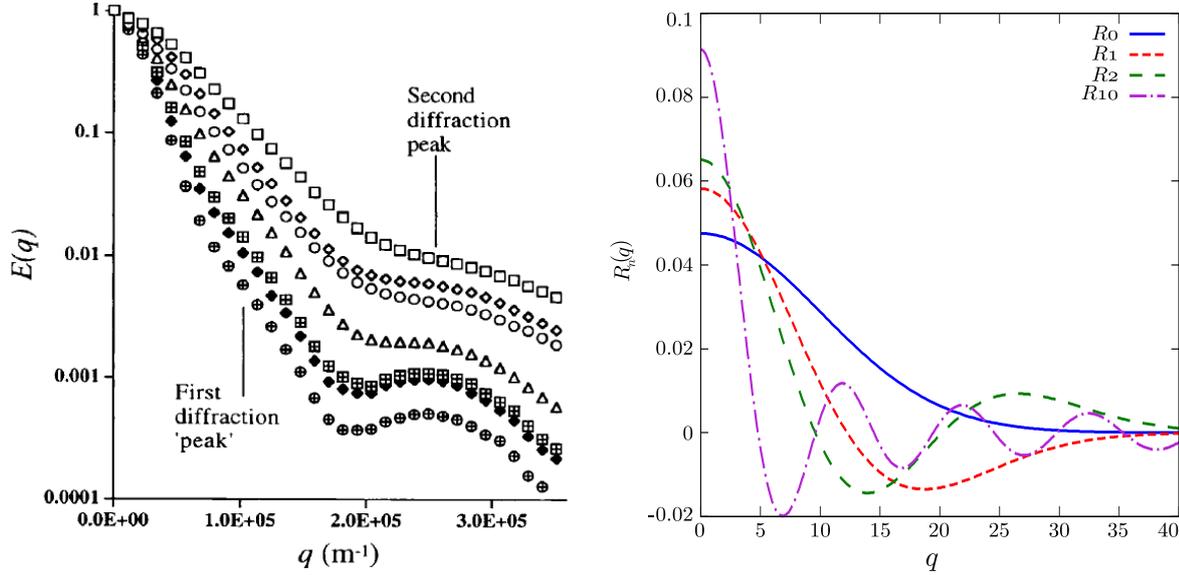


FIG. 7.2: Comparaison des courbes expérimentales du signal de diffusion radial (mesurées par Kuchel *et al.* [KUCHEL97]) (gauche) et de la base de reconstruction proposée (droite) R_n , pour $n = \{0, 1, 2, 3, 10\}$ et $\gamma = 100$.

7.3 Formalisme pour l'estimation de caractéristiques de la PDF

Une fois qu'une forme continue E du signal de diffusion a été estimé, nous considérons une caractéristique \mathcal{G} de la PDF, définie par projection au point $\mathbf{k} \in \mathbb{R}^3$, tel que

$$\mathcal{G}(\mathbf{k}) = \int_{\mathbf{p} \in \mathbb{R}^3} P(\mathbf{p}) H_k(\mathbf{p}) d\mathbf{p} \quad (7.7)$$

Nous montrons que le calcul de \mathcal{G} en passant par la transformée de Fourier de E n'est pas nécessaire. Au contraire, nous pouvons calculer \mathcal{G} directement à partir des coefficients de la base Ψ puisque celle-ci est orthonormée. La relation suivante est utilisée :

$$\mathcal{G} = \int P(\mathbf{p}) H_k(\mathbf{p}) d\mathbf{p} = \int E(\mathbf{q}) h_k(\mathbf{q}) = \sum_{nlm}^{\infty} a_{nlm} b_{nlm}^k \quad (7.8)$$

avec H_k la fonction de projection associée à \mathcal{G} et h_k la transformée de Fourier inverse de H_k . Les a_{nlm}^k et b_{nlm}^k sont respectivement les coefficients de E et h_k dans la base Ψ . L'intérêt de la

méthode proposée (7.8) réside dans sa rapidité et sa généralité pour le calcul d'une caractéristique quelconque de la *PDF*. Ce travail généralise donc les approches [DESCOT06, TOUCH99] et [ÖZAR06] avec les fonctions de projection respectives

$$FRT = H_k(\mathbf{p}) = 2\pi q' J_0 \left(2\pi q' \|\mathbf{p}\| \left(1 - \frac{\|\mathbf{p} \cdot \mathbf{k}\|}{\|\mathbf{p}\| \|\mathbf{k}\|} \right) \right)$$

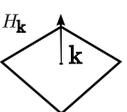
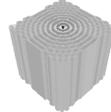
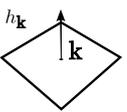
et

$$ISO = H_k(\mathbf{p}) = \delta(\|\mathbf{p} - \mathbf{k}\|) + \delta(\|\mathbf{p} + \mathbf{k}\|).$$

Nous pouvons même ici calculer la "véritable" ODF avec

$$H_k(\mathbf{p}) = \delta \left(1 - \frac{\|\mathbf{p} \cdot \mathbf{k}\|}{\|\mathbf{p}\| \|\mathbf{k}\|} \right)$$

(voir Tabl.7.1 pour plus de détails).

\mathcal{G}	ODF	FRT	ISO	P(0)
$H_k(\mathbf{p})$	$\delta \left(1 - \frac{\ \mathbf{p} \cdot \mathbf{k}\ }{\ \mathbf{p}\ \ \mathbf{k}\ } \right)$	$2\pi q' J_0(2\pi q' \ \mathbf{p}\ \left(1 - \frac{\ \mathbf{p} \cdot \mathbf{k}\ }{\ \mathbf{p}\ \ \mathbf{k}\ } \right))$	$\delta(\ \mathbf{p} - \mathbf{k}\) + \delta(\ \mathbf{p} + \mathbf{k}\)$	$\delta(\mathbf{p})$
				
$h_k(\mathbf{q})$	$\delta \left(\frac{\ \mathbf{q} \cdot \mathbf{k}\ }{\ \mathbf{q}\ \ \mathbf{k}\ } \right)$	$\delta(\ \mathbf{q}\ - \ \mathbf{q}'\) \delta \left(\frac{\ \mathbf{q} \cdot \mathbf{k}\ }{\ \mathbf{q}\ \ \mathbf{k}\ } \right)$	$\cos(2\pi \mathbf{q} \cdot \mathbf{k})$	$\frac{1}{Z^3}$
				

TAB. 7.1: Liste non exhaustive de caractéristiques de la *PDF* \mathcal{G} et de le fonction de projection H_k au point \mathbf{k} . Ces caractéristiques sont calculées comme une image volumique de taille $Z^3 = 64^3$ voxels et tracées en dessous de chaque formule. Etant donnée une direction \mathbf{k} , on illustre donc à quoi ressemble respectivement une caractéristique de la *PDF* et son expression dans l'espace q . Les deux premières figures de la première colonne (lignes 2 et 4) indiquent la direction de \mathbf{k} dans \mathbb{R}^3 .

En pratique, nous avons besoin de calculer h_k et b_{nlm}^k dans la base Ψ pour chaque vecteur $k \in \mathbb{R}^3$. Toutefois, ces calculs sont indépendants des données et ne sont requis qu'une seule fois. Les résultats, peu volumineux, peuvent donc être stockés dans la mémoire d'ordinateur pour un usage ultérieur. Il existe néanmoins une alternative encore plus rapide pour les caractéristiques angulaires ; *c-à-d.* $\mathbf{k} \in \mathcal{S}^2$ avec \mathcal{S}^2 le domaine de la sphère unité. On peut noter que toutes les méthodes basées sur une acquisition *HARDI* rentrent dans ce cadre. Dans ce cas, nous utilisons la propriété de rotation des harmoniques sphériques connues sous le nom de *théorème d'addition* : la rotation d'une harmonique sphérique de degré l peut être exprimée comme la combinaison linéaire d'harmoniques sphériques de même degré. En conséquence, la rotation d'un coefficient de la base Ψ peut être exprimé par :

$$\text{Rot}_{\alpha, \beta, \gamma} [a_{nlm}] = a'_{nlm} = \sum_{m'=-l}^l a_{nlm'} R_{mm'}^{(l)}(\alpha, \beta, \gamma) \quad (7.9)$$

avec a_{nlm} et a'_{nlm} respectivement un coefficient de la base Ψ et sa rotation. Nous définissons $R_{mm'}^{(l)}$ comme la matrice réelle de Wigner pour la rotation d'angle d'Euler (α, β, γ) en convention *zyz* [BLANCO97]. Ainsi, H_k ne doit être construit que pour $\mathbf{k} = \mathbf{z}$, les autres directions \mathbf{k} sont calculées avec l'Eq.(7.9). La Fig.7.3 récapitule les étapes nécessaires pour l'estimation d'une caractéristique \mathcal{G} définie sur la sphère unité.

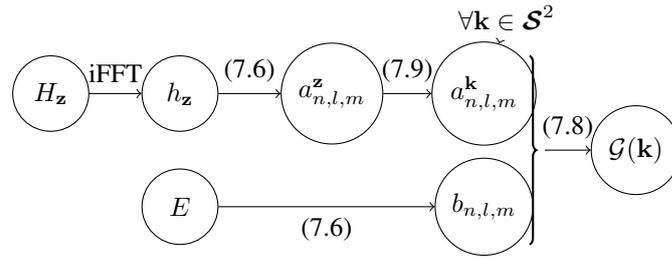


FIG. 7.3: Vue d'ensemble de l'algorithme pour l'estimation de caractéristiques \mathcal{G} de la *PDF* au point \mathbf{k} . $H_{\mathbf{k}}$ est la fonction de projection associée à \mathcal{G} et $h_{\mathbf{k}}$ est la transformée de Fourier inverse (iFFT) de $H_{\mathbf{k}}$. E est l'image de diffusion acquise.

7.4 Application et résultats

Des estimations de caractéristiques de diffusion sur des données *HARDI* et multi q -ball in vivo (Fig.7.4) ont été réalisés. Les données d'IRM de diffusion ont été acquises suivant deux sphères de 32 directions chacune. Les valeurs de gradients sont $b = \{0, 1000, 3000\} \text{ s/mm}^2$. Il y a ainsi un total de 65 images, acquises en 15 minutes. Le protocole d'imagerie parallèle SENSE a été utilisé avec un facteur d'accélération de 2 ; et près de 80% de l'espace k a été acquis. Les dimensions du volume image sont $112 \times 112 \times 60$ voxels, avec une résolution de $2 \times 2 \times 2 \text{ mm}^3$. Le temps de répétition $TR = 11490 \text{ ms}$ et le temps d'écho $TE = 85 \text{ ms}$. Le temps entre deux pulsations et temps des gradients de diffusions sont respectivement $\Delta = 42.2 \text{ ms}$ et $\delta = 26.3 \text{ ms}$. Les résultats de la Fig.7.4 ont été estimés avec des termes de troncature $N = 1$ et $L = 4$. Le temps de calcul total représente moins d'une minute sur un processeur 3 Ghz. Cela comprend la reconstruction continue du signal de diffusion dans la base Ψ Eq.(7.6). Cela comprend également le calcul de caractéristiques de la *PDF* Eq.(7.8,7.9) le long de 642 directions \mathbf{k} définies par un icosaèdre subdivisé pour les caractéristiques sphériques ou un seul point pour les caractéristiques scalaires.

Les Fig.7.4(a,b,c) illustrent les estimations calculées d'Anisotropie Fractionnelle (*FA*) [BAS-SER94] sur l'imagerie du tenseur DTI, Anisotropie Fractionnelle Généralisée [TUCH99] et de probabilité de non-déplacement $P(0)$ [WU07]. Quelques mesures sphériques de la littérature sont représentées comme référence (Fig.7.4(d,e,f)). La région concernée se situe près du corps calleux (*corpus callosum*), connue pour être le lieu de croisements de fibres. Contrairement à notre approche, ces méthodes requièrent une acquisition *HARDI* et ne tirent pas avantage de l'acquisition multi q -ball. Les résultats présentés Fig.7.4(d,e,f) sont donc estimés avec les données de la sphère $b = 3000 \text{ s/mm}^2$ uniquement.

De plus, il est intéressant de noter que chacune de ces caractéristiques peut se calculer ici dans un même formalisme Fig.7.4(h,i,j). Notre méthode reproduit avec succès les résultats des méthodes Q-Ball Imaging (QBI) et Diffusion Orientation Transform (DOT) (Fig.7.4(e,h) et Fig.7.4(f,i)). De plus, notre méthode permet aussi l'estimation précise de la "véritable" ODF (Fig.7.4(g)) et s'explique par : 1- la prise en compte de tous les échantillons du q -space qui permet d'améliorer sensiblement la qualité de l'estimation ; 2- l'*a-priori* de décroissance Gaussienne de notre base Ψ pour un ordre de troncature faible ($N = 1$).

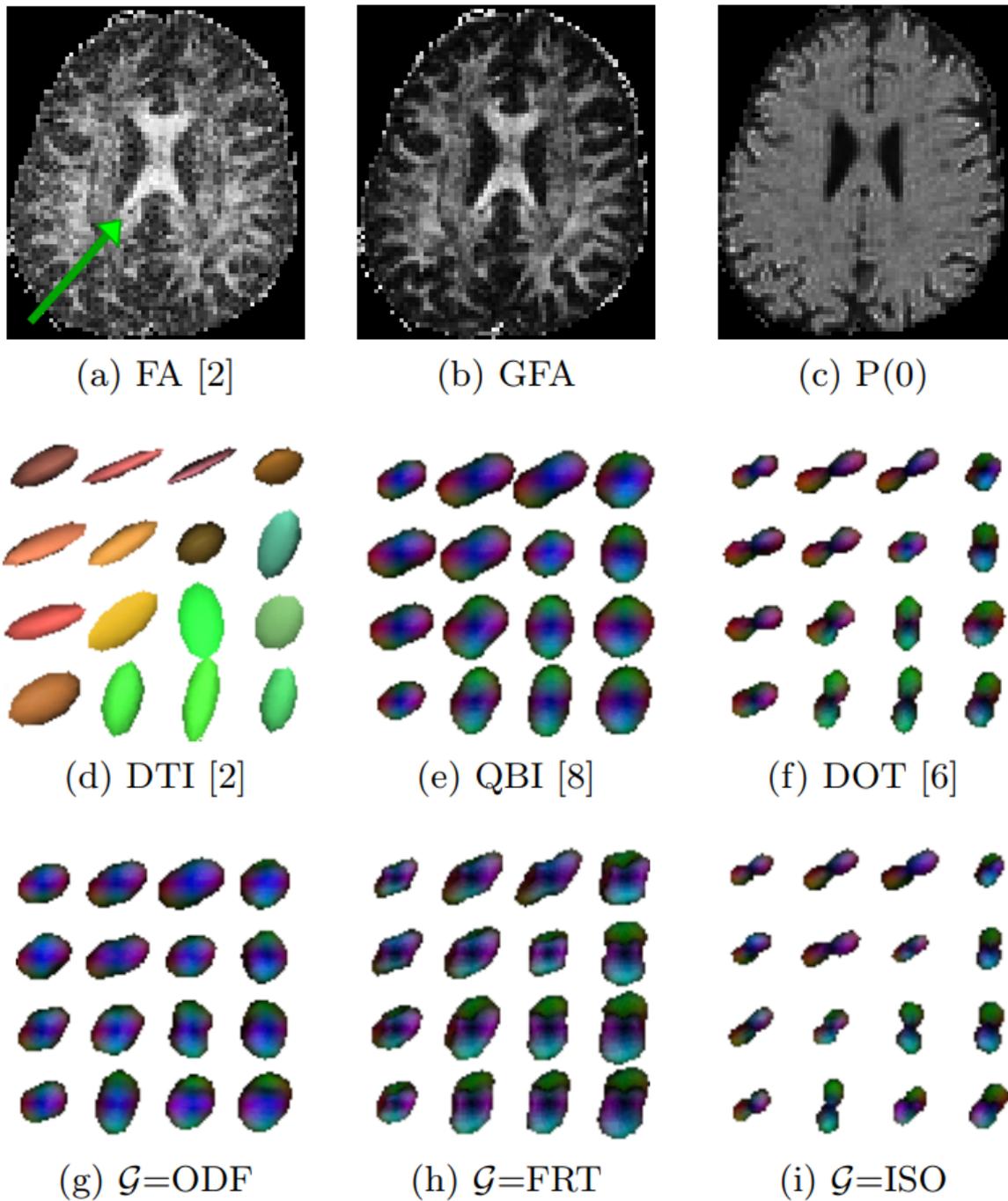


FIG. 7.4: Caractéristiques variées du tissu cérébral humain in vivo. 1ère ligne : caractéristiques scalaires. 2ème ligne : caractéristiques sphériques de la littérature. 3ème ligne : caractéristiques sphériques permises avec notre méthode.

7.5 Conclusion du chapitre

Dans ce chapitre, nous avons proposé une méthode en deux étapes pour l'estimation de caractéristiques des tissus cérébraux à partir de données d'IRM de diffusion. Le contexte clinique de cette modalité d'acquisition impose un nombre très réduit d'échantillons. Aussi dans une première étape, notre approche reconstruit le signal continu dans une base parcimonieuse. Ensuite, nous définissons une méthode de calcul rapide de caractéristiques de la *PDF*. Cette méthode unifie de nombreux travaux précédents et ouvre la voie pour la définition de nouvelles caractéristiques originales. Finalement, nous avons illustré la faisabilité de notre approche sur une acquisition in-vivo dans un intervalle de temps clinique raisonnable.

Ce chapitre montre aussi un exemple de limites du modèle tensoriel, pour un type d'application où le comportement à que l'on cherche à modéliser (ici la diffusion des molécules d'eau) suit une loi de probabilité plus complexe qu'une loi Gaussienne anisotrope.

Développements logiciels pour le traitement d'images

Sommaire

8.1	CImg : une bibliothèque C++ générique pour le traitement d'images	109
8.2	G'MIC : Un cadriciel multi-interfaces pour le traitement d'images	111
8.3	Expériences utilisateurs	116



Je présente ici mes activités principales de développement logiciel, menées au fil des années en parallèle à mes travaux de recherche et de publication. Ce travail annexe m'a permis, d'une part, d'accélérer grandement le temps d'implémentation des algorithmes présentés tout au long de ce manuscrit, en fournissant une base robuste et fonctionnelle d'opérateurs de traitements d'images sur lesquels s'appuyer, pour mes étudiants et moi-même. D'autre part, la diffusion des codes correspondants sous forme de bibliothèques et logiciels libres a été un moyen de partager de nombreux algorithmes avec la communauté scientifique du traitement d'images, mais aussi avec le grand-public, autorisant la reproductibilité scientifique des résultats obtenus, et la possibilité à tout à chacun de s'appropriier les différentes solutions algorithmiques proposées dans nos publications.

Période : 1999–2018.

Collaboration : Sébastien Fourey (MCF Ensicaen/GREYC, pour G'MIC), et plusieurs dizaines de contributeurs à travers le monde.

Publications/présentations associées :

- Conférences internationales : [C25] (IPOL'12).
 - Logiciels : [S4] (G'MIC), [S3] (CImg), [S2] (FISPEX), [S1] (GREYCstorage).
 - Présentations invitées : [T17] (JNCN'14), [T16] (LGM'14), [T15] (LGM'13), [T14] (FOSDEM'13), [T13] (Capitole du Libre'12), [T12] (Rencontres du Libre'12), [T9] (Workshop Thalès'09), [T4] (RMLL'07).
-

8.1 CImg : une bibliothèque C++ générique pour le traitement d'images

8.1.1 Problématique, méthodologie et solutions proposées

La recherche en traitement d'images demande des compétences à la fois en mathématiques appliquées, pour la conception de modèles et d'algorithmes, et des compétences en programmation informatique, car la résolution des problèmes considérés passe *in-fine* par une étape d'implémentation sur ordinateur. Or, les données images que l'on cherche à traiter sont de plus en plus complexes : volumineuses, multi-dimensionnelles, elles sont acquises avec des capteurs de plus en plus précis. Le temps où l'on traitait uniquement des images scalaires en 256 niveaux de gris est révolu. Aujourd'hui, la complexité des algorithmes et la diversité des images que l'on rencontre rendent d'autant plus difficile l'implémentation de méthodes *génériques* de traitement d'images sur ordinateur.

C'est en effectuant ce constat que j'ai commencé à développer (en C++) la bibliothèque *CImg*, dès 1999 et le début de ma thèse, effectuée dans l'équipe ODYSSEE de l'INRIA Sophia-Antipolis. Au cours des années, cette bibliothèque est devenue un projet relativement ambitieux, demandant un investissement continu, et des remises en questions permanentes de mes méthodes de développement.

L'idée a donc été d'élaborer une bibliothèque C++ minimale et légère qui définisse des classes les plus simples possibles pour la manipulation d'images et de séquences d'images multi-dimensionnelles (jusqu'à 3 dimensions spatiales) et multi-valuées (sans limitation sur le nombre de composantes), discrétisées sur des grilles cartésiennes. Après avoir rencontré un intérêt croissant pour cette bibliothèque dans l'équipe ODYSSEE de l'INRIA Sophia-Antipolis, j'ai cherché à l'étoffer, à mieux la structurer, et par la suite, à la diffuser (dès 2003).



FIG. 8.1: Logo actuel de la bibliothèque C++ libre *CImg*.

En 2004, lors de mon arrivée en tant que chargé de recherche CNRS au GREYC, j'ai pu très vite interagir avec *PANDORE*, la bibliothèque de traitement d'images alors développée et utilisée principalement dans l'équipe IMAGE. Les deux bibliothèques ayant des conceptions et des buts très différents, nous avons pu très logiquement continuer à les développer et les distribuer de manière indépendante.

Il existe aujourd'hui de nombreuses bibliothèques de traitement d'images disponibles sous licence libre. Mais, *CImg* a des atouts intéressants qui la distinguent de ses concurrentes : elle est relativement légère (environ *60kloc*), suffisamment générique (basée *template C++*), multi-plateforme, *thread-safe* et *multi-threadée*. Elle offre surtout un panel de fonctionnalités indispensables au traitement d'images : gestion des entrées/sorties (nombreux formats d'images

pris en compte), opérateurs classiques (filtrage, morpho-math, conversion d'espaces couleurs, transformations géométriques, fonctions de tracés, rendu 3D, analyse fréquentielle, calcul matriciel, ...). Elle est diffusée sous licence libre et permissive *CeCILL-C*, ce qui en fait une bibliothèque libre largement exploitable par les programmeurs C++.

Cette bibliothèque, dont je suis le coordinateur et développeur principal, est maintenant le fruit de collaborations avec quelques contributeurs extérieurs. J'estime néanmoins mon taux de contribution global à 95%. Son site web actuel est : <http://cimg.eu>.

8.1.2 Impact et statistiques d'utilisations

J'ai continué à développer activement et continuellement cette bibliothèque, en me focalisant ces dernières années plus particulièrement sur les aspects *thread-safe* et *multi-threadée*, notamment en privilégiant l'utilisation d'*OpenMP*.

Cimg, aujourd'hui rendue à sa version 2.2.2 (Mai 2018), est devenue une bibliothèque libre de traitement d'images ayant une visibilité relativement importante dans la communauté conjointe C++ / *Traitement d'images*. Cette bibliothèque dispose d'une communauté d'utilisateurs actifs de tous pays (entre 900 et 1000 visites par jour sur le site web, en moyenne), et de nombreux projets open-source d'importance l'utilisent en partie (voir la liste des projets recensés : <http://cimg.eu/links.shtml>). Plus de 350.000 téléchargements ont déjà été comptabilisés depuis sa mise en ligne (en 2003), avec une moyenne actuelle d'environ 50-70 téléchargements par jour (par IP unique). Un graphique des téléchargements sur une fenêtre de 200 jours (fin 2017-début 2018) est visible sur la Fig.8.2 ci-dessous.

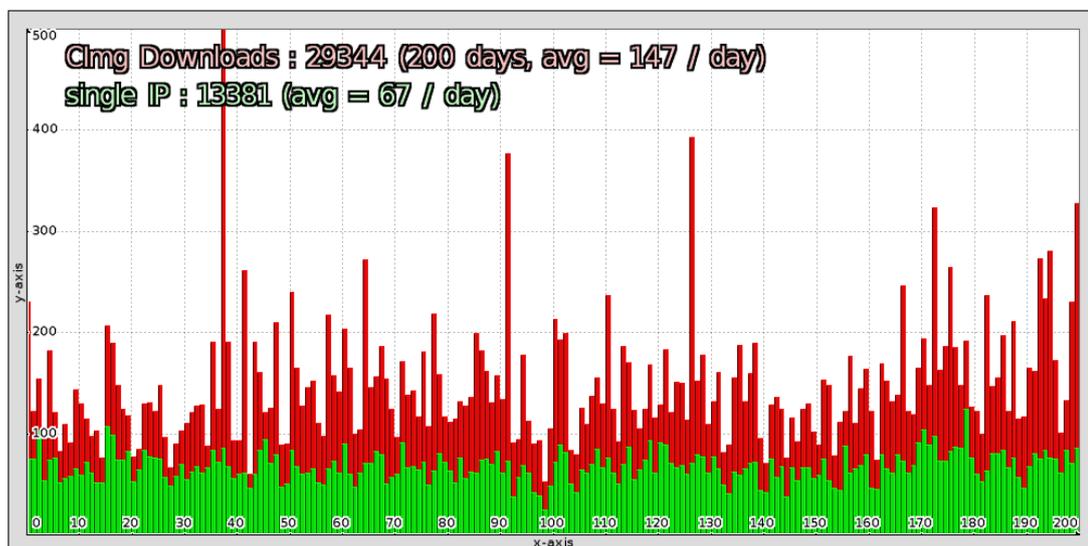


FIG. 8.2: Statistiques des téléchargements de *Cimg*, sur une fenêtre de 200 jours (fin 2017-début 2018).

En rouge : Nombre de téléchargements total comptabilisé, chaque jour.

En vert : Nombre de téléchargements comptabilisé par IP différente, chaque jour.

Ces chiffres sont loin d'être négligeables, en regard de la communauté d'utilisateurs potentiellement concernés (programmeurs C++ seulement, dans le domaine du traitement d'images).

Localement, *CImg* permet à mes collègues, aux étudiants que j'encadre, et à moi-même, de disposer rapidement d'un cadre de travail complet pour aborder des problèmes de traitement d'images variés, et c'est également le cas dans d'autres équipes de recherche à travers le monde. Le développement de *CImg* s'est révélé très (très) chronophage dans le passé (10 à 12 heures hebdomadaire), mais le temps que j'y consacre aujourd'hui est plus réduit (environ 1 à 2 heures hebdomadaire), la bibliothèque étant aujourd'hui considérée comme "stable", sans projets d'évolutions majeures prévues.

Bien conscient qu'il est extrêmement difficile d'évaluer l'impact *réel* d'un logiciel (et encore plus d'une *bibliothèque* de programmation !), même avec toutes les statistiques possibles et imaginables, j'ai décidé de joindre en fin de chapitre, des avis d'utilisateurs satisfaits, attestant de l'utilité de *CImg* tant pour la recherche que pour l'enseignement en traitement d'images. J'ai fait de même pour le logiciel *G'MIC*, présenté en section suivante.

8.2 G'MIC : Un cadriciel multi-interfaces pour le traitement d'images

8.2.1 Problématique, méthodologie et solution proposée

En m'intégrant dans une équipe de recherche telle que l'équipe IMAGE du GREYC, dédiée au traitement d'images d'un point de vue très *générique*, j'ai vite été confronté à des problématiques diverses, impliquant des modalités d'images variées (photos couleurs, volumes IRM 3D+t, échographies, images de microscopie, images scanner, images satellites, etc.). Dans un premier temps, on souhaite souvent *explorer* ces images, c-à-d les visualiser, mais aussi appliquer quelques traitements préliminaires de base, pour mieux comprendre leur structure et les informations essentielles qu'elles contiennent.

Malheureusement, il existe bien peu d'outils libres "tout-en-un" permettant de lancer des traitements (même classiques) sur des données images aussi génériques. En particulier si l'on cherche à traiter des images "non conventionnelles" (c-à-d hors des images 2D couleurs classiques), comme par exemple des images volumiques multi-canaux ou des images à valeurs flottantes, on se trouve très vite confronté à l'utilisation de logiciels multiples (et encore, quand ils existent !) pas toujours compatibles entre eux, et/ou très difficilement "scriptables" pour du traitement par lot. Il n'est pas rare que l'on doive alors développer des outils *ad-hoc*, prototypés rapidement et peu maintenables dans la durée.

En corollaire, arriver à diffuser des algorithmes prototypés dans l'équipe pour les rendre immédiatement et facilement disponibles pour les chercheurs ou le grand public est difficile et coûteux en temps. Cela nécessite le développement d'interfaces graphiques dédiées et/ou l'empaquetage des logiciels, travail rébarbatif s'il en est. Ces développements additionnels deviennent vite chronophages et j'ai pu remarquer que la plupart des équipes de recherche n'ont simplement pas les ressources humaines nécessaires pour les réaliser et/ou les rendre pérennes.

C'est en faisant ce constat que j'ai initié en 2008, l'écriture du logiciel *G'MIC* (pour *GREYC's Magic for Image Computing*, acronyme inverse de *CImg*). Initialement élaboré comme un outil de traitement d'images génériques 1D/2D/3D multi-valuées se focalisant sur une utilisation *en ligne de commande*, il a évolué peu à peu pour devenir un *cadriciel (framework)* à part entière, définissant un interpréteur de *langage de script* dédié au traitement d'images génériques.

L'idée principale du projet est d'utiliser cet interpréteur dans plusieurs interfaces différentes d'utilisation. Chaque nouvelle fonctionnalité de traitement développée, ajoutée à l'interpréteur, devient *de facto* immédiatement disponible sur toutes les interfaces à la fois. En factorisant ainsi ces différents types d'interactions possibles, on minimise le travail nécessaire à la maintenance de la fonctionnalité ajoutée. En élaborant des interfaces suffisamment diverses, on peut s'adapter à plusieurs publics, et la diffusion des algorithmes scientifiques que nous développons dans l'équipe devient alors bien plus facile à réaliser. Depuis 2008, nous avons développé les interfaces d'utilisation suivantes :

1. **Ligne de commande** : *gmic* est un outil que l'on peut invoquer à partir de la ligne de commande. De ce point de vue, on peut le voir comme un concurrent de la suite d'outils libres bien connue *ImageMagick* (<http://www.imagemagick.org>). J'en suis le développeur principal.
2. **Greffons pour GIMP et Krita** : *gmic_gimp* est un greffon (*plug-in*) développé pour les logiciels de retouche d'image libres et grand public *GIMP* (<http://www.gimp.org>) et *Krita* (<http://www.krita.org>). J'étais le développeur principal du greffon, avant sa refonte complète en 2017, par *Sébastien Fourey* (MCF équipe IMAGE) qui en est aujourd'hui le mainteneur principal. C'est actuellement l'interface la plus populaire du projet *G'MIC*.
3. **Bibliothèque C++** : *libgmic* est une bibliothèque C++ libre pouvant être intégrée dans un logiciel tiers pour profiter de toutes les possibilités de traitement qu'offre *G'MIC*. Je suis en contact avec des développeurs de divers logiciels libres (*Krita*, *EKD*, *Photoflow*, *Flowblade*, *Veejay*) pour aider à l'intégration de *libgmic* dans leur logiciel. J'en suis le développeur principal.
4. **Plateforme de démonstration** : *ZArt* est une interface Qt permettant de manipuler (via *G'MIC*) des images provenant de la webcam ou de fichiers vidéos. Cette interface est utilisée par exemple lors des *Fêtes de la Science* pour réaliser des démonstrations interactives autour du traitement d'images. Cette interface a été développée principalement par *Sébastien Fourey* (MCF équipe IMAGE).
5. **Service web** : *G'MIC Online* est un site web, disponible à l'adresse : <https://gmicol.greyc.fr/>. Ce service web permet à quiconque de tester différents filtres et effets de traitement d'images couleurs, directement à partir d'un navigateur web sans installation préalable de logiciel. Cette interface a été développée principalement par *Sébastien Fourey* (MCF équipe IMAGE) et *Christophe Couronne* (ingénieur développement du GREYC).

G'MIC est bien sûr développé en se basant principalement sur les fonctionnalités de la bibliothèque *CImg*, et sait tout aussi bien traiter les images 2D couleur classiques, les images volumétriques 3D multi-canaux, que les séquences d'images (elles-mêmes pouvant être volumiques), et les formats d'images à grande profondeur de bits (16bits ou 32bits flottant par canal et par pixel). Il comprend un ensemble d'opérateurs élémentaires d'images pour le filtrage, la transformation géométrique, les opérations arithmétiques, etc.. Plus de *1000 opérateurs* de traitement sont aujourd'hui disponibles dans le logiciel. Ces opérateurs peuvent se combiner avec simplicité et flexibilité et la création de nouveaux opérateurs est donc facilitée, et rend le développement du logiciel et le prototypage de nouveaux algorithmes de plus en plus rapide. *G'MIC* possède des modules de visualisation simples mais puissants, et la possibilité d'extraire et de visualiser des caractéristiques 3D des images (cartes d'élévation, isophotes, isosurfaces, ...). Il

est également distribué en suivant un modèle de diffusion libre (licences *CeCILL-CeCILL-C*). Le site web du projet est actuellement à l'adresse : <http://gmic.eu/>.



FIG. 8.3: Logo et mascotte actuels du logiciel libre *G'MIC*.

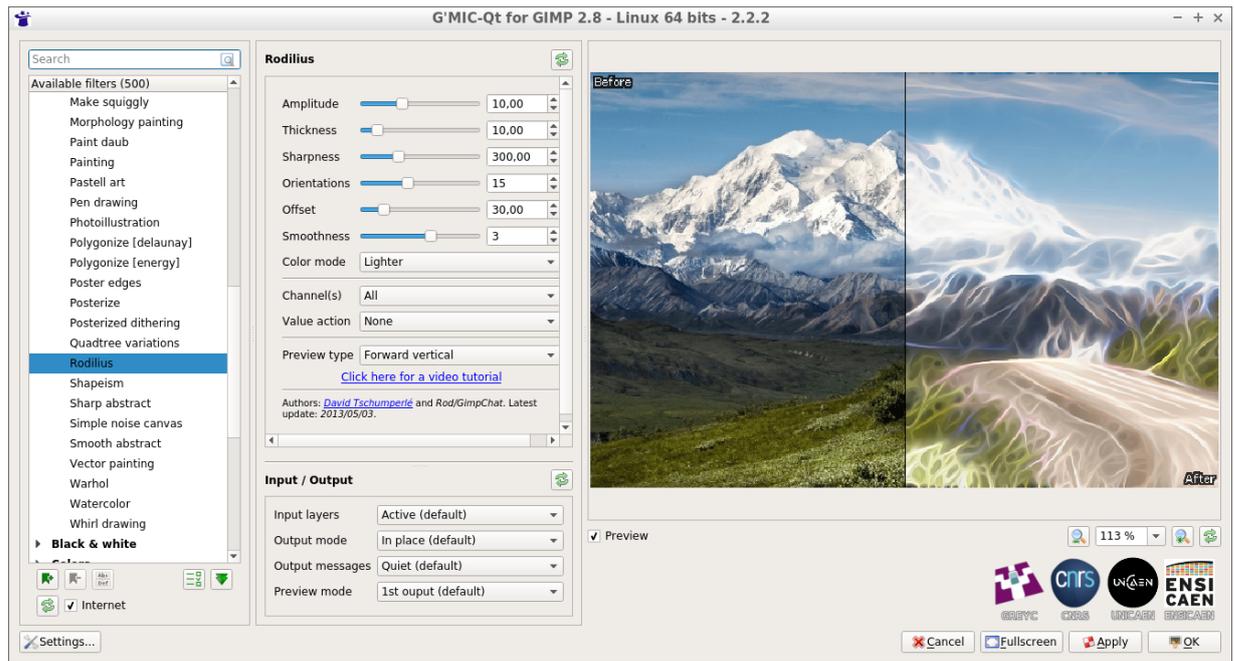
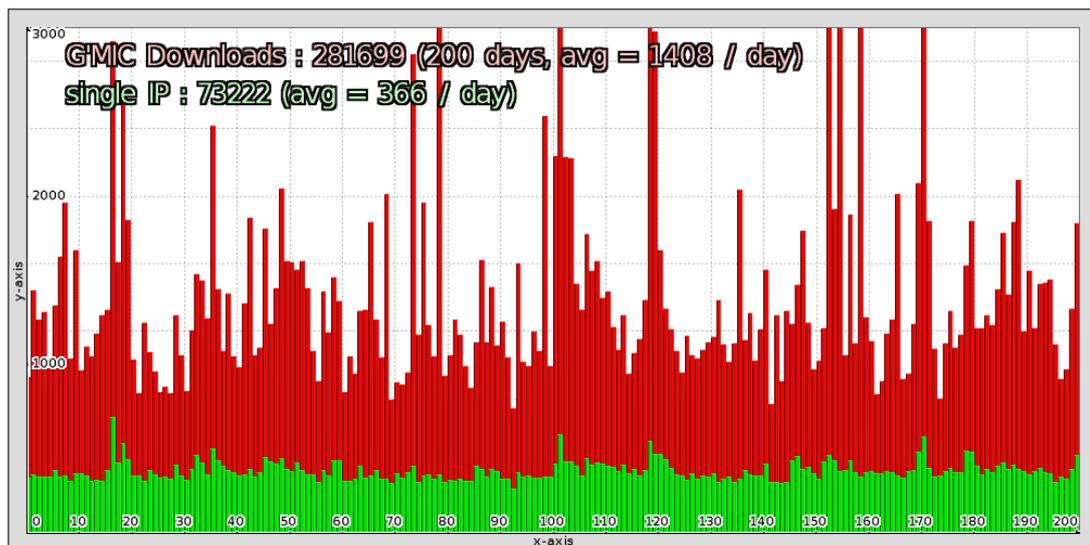
Ce logiciel, dont je suis le coordinateur et le développeur principal, est maintenant le fruit de collaborations avec plusieurs contributeurs extérieurs. J'estime mon taux de contribution global à 90%.

8.2.2 Impact et statistiques d'utilisations

Ces dix dernières années ont vu la popularité du logiciel *G'MIC* grandir considérablement, notamment grâce à la mise à disposition des greffons d'abord pour *GIMP*, puis pour *Krita*, deux logiciels libres très populaires de dessin numérique et de retouche d'images. Ces greffons ont amené un grand nombre d'utilisateurs nouveaux à *G'MIC*. Aujourd'hui, notre greffon est plébiscité comme l'un des meilleurs disponibles pour *GIMP*, de par la diversité et la complexité des algorithmes de traitement d'images qu'il propose (environ 500 *filtres et effets* variés, disponibles pour le grand public).

Quelques chiffres significatifs, mesurés en Février 2018 :

- Une recherche “G'MIC” sur *Google* retourne plus de 180,000 résultats.
- Le nombre de visites sur le site web du projet oscille entre 900 et 1300 par jour (en hausse, depuis 2008), ce qui correspond à environ 400-500 visiteurs quotidiens uniques.
- Le nombre total de téléchargements du logiciel a dépassé la barre des 3.5 millions en 2017. A noter qu'il nous est de plus en plus difficile de comptabiliser ces téléchargements car nous n'avons pas accès aux statistiques de téléchargements provenant des plateformes extérieures au site web du projet (téléchargements réalisés en passant par les systèmes de paquets, sous *Linux* par exemple). Sur le site, on dénombre entre 350 et 400 téléchargements quotidiens du logiciel (par IP unique, voir Fig.8.5). Il est probable cependant que le chiffre réel soit bien plus important, en supposant que l'on soit capable d'avoir accès aux statistiques de toutes les sources possibles de téléchargements proposées pour ce logiciel.
- Il est évidemment difficile de relier le nombre d'utilisateurs journaliers du logiciel avec le nombre de téléchargements mesurés. Nous avons cependant accès à une statistique supplémentaire intéressante : le greffon possède en effet un système de mise à jour automatique de ses filtres d'images. Ce système se déclenche par défaut une fois par semaine (ou lorsque l'utilisateur le demande explicitement). Or, chaque demande de mise à jour peut

FIG. 8.4: Le greffon *G'MIC* pour GIMP, en action.FIG. 8.5: Statistiques des téléchargements de *G'MIC*, sur une fenêtre de 200 jours (fin 2017-début 2018).

En rouge : Nombre de téléchargements total comptabilisé, chaque jour.

En vert : Nombre de téléchargements comptabilisé par IP différente, chaque jour.

être comptabilisée puisque la requête réseau correspondante est logguée par le serveur principal où est hébergé le logiciel. Ce nombre de requêtes oscille entre 450 et 600 par jour (par IP unique), qui correspond donc au nombre *minimum* d'utilisateurs journaliers du logiciel. (voir Fig.8.6). Le nombre journalier réel d'utilisateurs est probablement bien plus élevé, puisque la majorité des utilisateurs ne rentrent pas dans ce comptage (la fréquence moyenne de mise à jour du logiciel n'est pas journalière !). On suppose donc que

le logiciel *G'MIC* est plutôt utilisé de manière *journalière* par 1000 – 2500 personnes à travers le monde, ce qui en fait un logiciel très utilisé.

- Le nombre d'abonnés aux flux d'informations liés au logiciel est de 2284 (flux principal, *Google+*), 708 (flux secondaire, *Twitter*), et 1086 (chaîne vidéo, sur *Youtube*).
- Des communautés actives d'utilisateurs se sont formées, sur des sites extérieurs, par exemple <http://pixls.us>, <http://gimpchat.com>, ou encore par le biais d'une communauté *Google+*.
- Le site <http://openhub.net>, qui réalise des estimations automatiques de coûts de développements à partir des sources de projets libres, estime le coût global de développement du couple *CImg* + *G'MIC* à 3,300,000\$, avec un effort de développement correspondant à 8 *personnes-années* (pour *G'MIC*) et 50 *personnes-années* (pour *CImg*). Ces chiffres sont évidemment à modérer, mais donnent néanmoins une idée de l'effort constant de développement logiciel que je fournis depuis 2003 (date de la première mise en ligne de *CImg*).

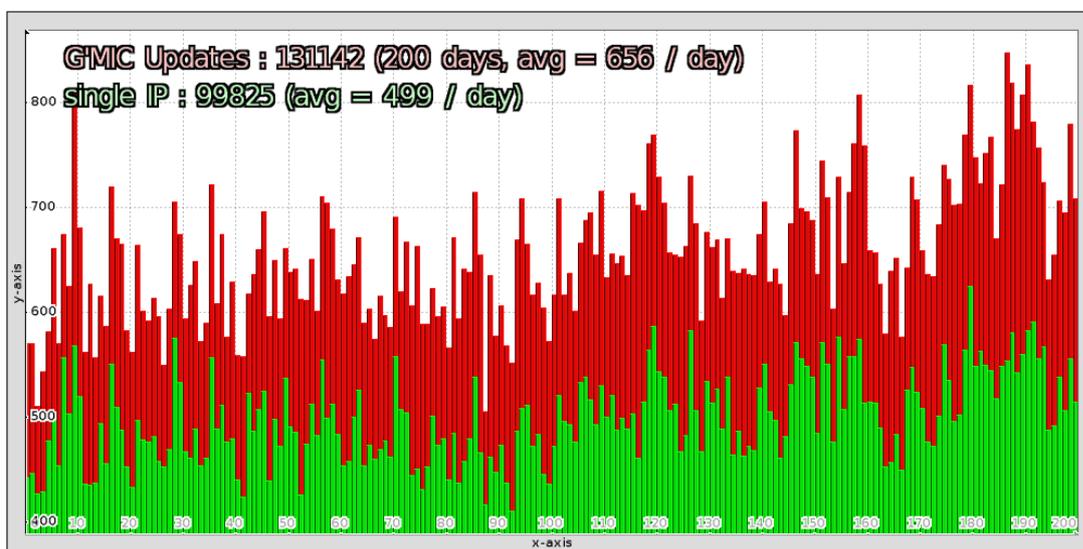


FIG. 8.6: Statistiques des demandes de mises à jours automatiques de *G'MIC*, sur une fenêtre de 200 jours (fin 2017-début 2018).

En rouge : Nombre de requêtes total comptabilisé, chaque jour.

En vert : Nombre de requêtes comptabilisé par IP différente, chaque jour.

G'MIC est devenu un projet de plus en plus ambitieux (et manifestement utile à beaucoup de gens), et qui m'a donc demandé un nombre d'heures de travail considérable durant ces dix dernières années (environ 15-20 heures hebdomadaires en moyenne, pour un programme de plus de 90.0000 lignes de code). C'est un logiciel que j'utilise au quotidien pour faciliter mon travail de recherche, notamment pour l'exploration de données et l'aide au prototypage de nouveaux algorithmes. Plusieurs de mes collègues l'utilisent régulièrement, y compris dans d'autres laboratoires. En fin de chapitre, j'ai rassemblé quelques lettres de témoignages d'utilisateurs de *G'MIC* aux profils variés (académiques, indépendants, institutionnels) et d'horizons divers qui attestent de l'utilité de ce logiciel. Devant l'engouement rencontré par ce projet, je m'efforce de le documenter et de le valoriser au maximum, pour attirer encore plus d'utilisateurs et de contributeurs, et en faire un vitrine logicielle pour le laboratoire GREYC :

- J'ai écrit et je mets à jour régulièrement un manuel de référence [26] (448 pages à ce jour), décrivant l'utilisation du cadriciel *G'MIC* et l'ensemble des commandes de traitement d'images disponibles.
- Je passe du temps à enrichir le logiciel et essayer de faire vivre la communauté d'utilisateurs, via l'écriture de pages web, la réponse aux questions sur les forums, l'aide aux nouveaux contributeurs, etc.
- J'ai présenté *G'MIC* dans diverses conférences majeures du logiciel/graphisme libre : *RMLL*, *FOSDEM*, *Libre Graphics Meeting*.
- J'essaie de valoriser ce développement logiciel au niveau des différentes tutelles du GREYC. Ainsi, le rapport d'activité national 2013 du CNRS le mentionne (page 45) Le rapport d'évaluation quinquennal du GREYC par le comité HCERES en fait également mention dans les points forts du laboratoire. Je suis en contact avec le service valorisation du CNRS (délégation Normandie) pour définir des pistes de valorisation propres au logiciel libre (financements participatifs, dons, montage de consortium, etc.).
- J'écris et je publie des articles de vulgarisation "en ligne" à propos de *G'MIC*, notamment sur des sites d'information grand public autour du logiciel libre (*Linuxfr* en particulier, voir une liste d'articles rédigés ici : <http://linuxfr.org/tags/gmic/public>).

Cette activité autour du logiciel libre a représenté une part importante de mon activité, et mérite à mon sens d'apparaître comme une de mes contributions significatives (et d'avoir donc son chapitre dédié dans ce manuscrit). Par le biais de ce logiciel, je suis en mesure d'interagir aujourd'hui avec de nombreux artistes amateurs et professionnels aux profils très différents (photographes, illustrateurs, peintres numériques, vidéastes, ...). Dans leur activité, ces personnes ont régulièrement des problématiques spécifiques et non triviales de traitement d'images, dont certaines constituent à elles-seules des sujets de recherche originaux, intéressants et non totalement résolus (transfert de style ou de couleurs, colorisation automatique ou semi-automatique, ...). Avoir réussi à fédérer une large communauté d'experts infographistes autour de notre logiciel *G'MIC* est une chance, dont je vais essayer de profiter pour mes collaborations et activités futures. Ceci sera détaillé dans le chapitre 9.

8.3 Expériences utilisateurs

Les pages suivantes compilent un ensemble de témoignages de quelques utilisateurs de mes logiciels et bibliothèques libres *CImg* et *G'MIC*. Ces témoignages cherchent à montrer d'une part, l'utilité de ces développements logiciels pour la communauté du traitement d'images (aussi bien du point de vue scientifique qu'artistique), et d'autre part, le caractère générique de ces implémentations informatiques, illustré par la grande variété des profils d'utilisateurs et des applications auxquelles ils s'intéressent.

December 21, 2015

David Tschumperlé
GREYC (UMR-CNRS 6072)
6 Bd du Marechal Juin
14050 Caen Cedex / France

To whom it may concern,

It is my pleasure to write a letter in support of the development of the image processing library CImg.

My responsibilities in *Cambridge Advanced Imaging Centre* span microscopy image analysis and data management. The particularity of the microscopy imaging in our centre is the high amount of fluorescence data collected per experiment (e.g. embryogenesis recorded over several days via light sheet microscopy resulting in one or several terabytess of data). Moreover, the scientific environment requires rapid development of prototypes and/or full analysis tools. However traditional scripting languages, such as Matlab, struggle to cope (in reasonable amount of time) with the amount of data that needs to be processed.

The CImg library is a unique software that allows to take full advantage of the C++ language, offers an intuitive and easy-to-use image processing library that includes many cutting edge algorithms implemented by the CImg community and via pre-compiled code it can be easily integrated in Matlab, such that rapid prototyping and advanced programming techniques are available at the same time.

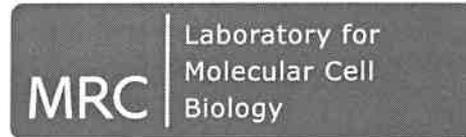
I intend to present and recommend this software at the Light Microscopy Facility Managers Meeting on 7-8 January 2016 in Oxford, UK. I believe that by combining CImg with Bio-formats (an image importing software for several microscopy formats) one has an ideal tool for analysis of big microscopy data, of huge benefit for the microscopy community.

Sincerely,



Leila Mureşan

Jérôme Boulanger
MRC Laboratory of Molecular Biology
Francis Crick Avenue, Cambridge Biomedical Campus
Cambridge CB2 0QH, UK
Téléphone: (+44)-16



A qui de droit,

18 décembre 2015

Madame, Monsieur,

Chargé de recherche au CNRS, à l'Institut Curie et désormais en détachement au MRC-LMB à Cambridge, je consacre mes travaux au développement d'algorithmes pour le traitement d'images pour les données de microscopie. Je m'intéresse en particulier au domaine de la microscopie photonique et son application à la compréhension de mécanismes moléculaires en biologie cellulaire. Dans ce contexte, je développe des prototypes logiciels me permettant de valider les méthodes développées. Certains de ceux-ci sont alors amenés à devenir de véritables outils pour les biologistes.

Mon développement logiciel est essentiellement basé sur la bibliothèque de traitement d'images CImg développée par David Tschumperlé. Elle permet un prototypage rapide tout en gardant des temps d'exécution raisonnables. Récemment, j'ai également fait appel à l'interpréteur G'MIC qui permet de tester encore plus rapidement des algorithmes mais aussi d'explorer de façon interactive les données de microscopie.

Nous avons en particulier développé dans [2] une méthode de dé-bruitage de séquences d'images 3D pour la microscopie de fluorescence. Le logiciel basé sur la bibliothèque CImg est maintenant utilisée dans de nombreux laboratoires de biologie cellulaire et une licence a été commercialisée par l'INRIA et l'INRA auprès de la société Roper Scientific.

Au cours d'une autre étude sur la reconstruction d'images 3D acquises en microscopie par ondes évanescentes [1], j'ai mis en œuvre l'algorithme de reconstruction encore une fois en utilisant la librairie CIMG. J'ai aussi utilisé l'interpréteur G'MIC pour définir des chaînes de traitements afin de réaliser les figures de l'article.

Veuillez agréer l'expression de mes plus sincères salutations,

Jérôme Boulanger

A handwritten signature in black ink that reads 'J Boulanger'. The signature is written in a cursive style with a horizontal line underneath the name.

- [1] Jérôme BOULANGER et al. "Fast high-resolution 3D total internal reflection fluorescence microscopy by incidence angle scanning and azimuthal averaging". In : *Proceedings of the National Academy of Sciences* 111.48 (2 déc. 2014), p. 17164–17169.
- [2] Jérôme BOULANGER et al. "Patch-based nonlocal functional for denoising fluorescence microscopy image sequences". In : *IEEE Transactions on Medical Imaging* 29.2 (fév. 2010), p. 442–454.



DIRECTION GÉNÉRALE DES DOUANES
ET DROITS INDIRECTS

SERVICE NATIONAL DE DOUANE JUDICIAIRE
2, mail Monique Maunoury (TSA 10314)
94853 IVRY-SUR-SEINE Cedex
Site Internet : www.douane.gouv.fr

IVRY-SUR-SEINE, LE 23 DÉCEMBRE 2015

La magistrate, sous-directrice déléguée
aux missions judiciaires de la douane,
chef du Service National de Douane Judiciaire

À

M. Abderrahim EL MOATAZ BILLAH
Equipe Image – GREYC
Université de Caen - ENSICAEN
6 Boulevard du Maréchal Juin
CS 45053
14050 CAEN cedex 4

n° de référence
15000 698

Objet : Remerciements
Réf. : Outil de retouche d'image pour l'analyse bancaire G'MIC développé
pour le SNDJ

Monsieur,

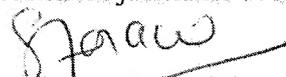
mon unité d'analystes a pris contact avec vous en septembre 2015 pour vous soumettre une problématique de retrait non-destructif d'une mention récurrente sur des fichiers pdf destinés à être exploités durant les enquêtes financières dont est saisie la douane judiciaire. Ces filigranes empêchent le traitement automatisé de ceux-ci et rendent obligatoire une saisie manuelle des données qu'ils contiennent.

A ce titre, la disponibilité et la compétence dont vous et vos chercheurs, et en particulier M. David TSCHUMPERLÉ, avez fait preuve sont à souligner.

Celui-ci nous a livré en novembre un outil performant, personnalisé et facile d'emploi, sous forme d'un greffon au logiciel GIMP, qui est immédiatement entré en production au service et fait gagner un temps précieux aux enquêteurs en matière d'analyse bancaire.

Je vous adresse donc à vous, à M. TSCHUMPERLÉ et à l'ensemble de votre équipe, mes remerciements les plus sincères.

La sous-directrice,
déléguée aux missions judiciaires de la douane


Solange MORACCHINI



January 29, 2016

Gentlepeople,

I first encountered G'MIC in early 2012. At the time, I was using ImageMagick to generate special effects footage, this to be composited with 3D animation in various ways.

The technical results were satisfactory. However, I was melancholic about long-term project management. My unease was rooted in the nature of the ImageMagick command line. It seemed to have a complex syntax which varied from effect to effect, apparently according to the author's fancy in working out some facet of the project. I was not confident that the package could extend to new effects without a significant investment of additional time.

About that time, G'MIC came to me, "as is," in a Fedora distribution and would have escaped my attention but that it billed itself as a "friendly competitor" of ImageMagick, which piqued my interest. After some experimentation with G'MIC, here is what came to matter for me:

1. G'MIC, like ImageMagick, is computationally "deep," calculating at 32 bit floating point resolutions. This is important in animation as insufficient computational precision leads to "pops" and "flashes."
2. G'MIC has a fundamental Image data structure that intrinsically encompasses a third dimension, which could be readily interpreted as either a volumetric or temporal axis. Naturally, my interests run to the temporal interpretation. The design of nearly all commands intrinsically adopt this structure, so that, for example, the blurring of a single pixel spreads readily into the temporal as well as the spatial dimensions, readily achieving a "pop" animation effect that, in less accommodating settings, would entail much tedious work.
3. G'MIC has a very consistent command line notation. This feature lends itself to straightforward extension, in that pipelines can be readily converted into self-contained commands and these may go forth as building blocks to even more ambitious pipelines. This makes package and script maintenance very straightforward.

For clients and colleagues somewhat adverse to command lines and shells, there is the G'MIC-Gimp plug-in that let me wrap a little bit of point-and-click user interface around scripts. While that was not a prime concern for me at the time, it has since turned out to be nice icing on the cake.

From a technical perspective, the results of G'MIC are on par with commercial offerings. Commercial offerings – primarily Photoshop or After-Effects plug-ins – are precoded, fixed entities that offer little opportunity for extension and adaptation. Many of them are very good, but all are more-or-less frozen bits of computational intelligence, with little promise for further adaptation. In contrast, the G'MIC

framework is entirely flexible, and its study rewards one with a large degree of adaptability and extendibility of effects. One effect, with a little parametric tweaking, becomes a new effect, obtained almost for free.

I should like to point out that these key advantages became manifest well before I made any acquaintances with David Tschumperlé. G'MIC entirely sold itself.

When I did make my acquaintance with David Tschumperlé, his enthusiasm – not only for G'MIC, but the overarching discipline of computer vision – left me astounded. He works very quickly, and yet retains the precision and care that haste often obliterates. David's enthusiasms are not just for G'MIC – as the program's author, one could expect that – but for the craft of envisioning phenomena computationally. One discovers that while G'MIC is quite handy at working out animation special effects, its true setting is that of an exploratory tool, one invented by a person deeply concerned with and fascinated with first principles. In this context, one understands the wellspring of G'MIC's flexibility and command line consistency. It is a tool intended for the toolmaker's exploration, to assist in that exploration, and to distract as little as possible with maintenance and overhead.

In closing, I should like to remark upon one aspect of David Tschumperlé's work which perhaps does not have a technical aspect, but, I think, is quite important.

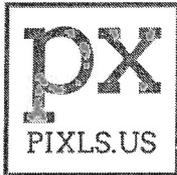
America's view of the world in the matter of software is very much akin to illustrator Saul Steinberg's celebrated and much-parodied *New Yorker* Magazine cover for March 29, 1976 ("*View of the World from Ninth Avenue*"). Largely, it is one of "If it is not written here, who's to know?" Europe – whatever happens in Europe – it is one of those lumps on the far horizon. Perhaps it is behind China. We Americans understand that you have computers too. We hear that you even make pretty pictures with them.

I am being facetious, of course, but not by much. David Tschumperlé's efforts with G'MIC in large measure dispel that terrible attitude. His work showcases high quality software of European – and French! – origins. Perhaps being an ambassador was not one of his aims, but in point of fact he has become one – and a very important one. That may be, in the long run, the most important result of all of his efforts. In the very least, I can speak for one American who, having found an especially nice gem of European origins, is now very much inclined to look for more.

Regards,

A handwritten signature in black ink, appearing to read "Garry R. Osgood". The signature is stylized, with the first name "Garry" written in a cursive-like script and "R. Osgood" in a more blocky, uppercase style.

Garry R. Osgood



PIXLS.US

Patrick David

~~XXXXXXXXXXXX~~
~~XXXXXXXXXXXX~~
Mobile, AL 36608 USA

Phone: ~~(504) 222-XXXX~~

E-mail: ~~XXXXXXXXXXXX@XXXXXX~~

URL: <https://pixls.us>

January 5, 2016

GREYC (UMR-CNRS 6072)
Equipe IMAGE
6 Bd du Marechal Juin
14050 Caen Cedex / France

To whom it may concern,

My name is Patrick David, a photographer and artist working entirely with Free and Open Source Software (FOSS). I am the creator of the photography community and website at PIXLS.US, an active member of the GNU Image Manipulation Program (GIMP) team, and a collaborator on various Libre culture projects. I have been working with various Free Software packages over the years while mentoring and teaching new photographers, film makers, and visual artists.

I have been a user of G'MIC (previously as GREYCstoration for noise reduction) for many years. This image processing framework is an **essential** component in the tools for visual artists and photographers.

As a photographer the ability to quickly create filters and to have access to a powerful image processing framework has been essential. There are many types of algorithms used in a professional photographic workflow that are simply not available by any other means.

The open nature of the licensing and community around G'MIC/CIMG has allowed artists to share and improve on algorithms and filters for processing, greatly accelerating their use and adoption. I feel this is important as it allows anyone to learn, experiment, use, and collaborate on all manner of image processing options for scientific, as well as artistic, applications. In many cases, the results being created are far ahead of any proprietary implementation.

G'MIC is an amazing and essential resource for photographers and visual artists that is unique, empowering, and simply irreplaceable.

Best regards,

A handwritten signature in black ink, appearing to be 'Patrick David', with a long horizontal line extending to the right.

Patrick David



Steve Gillow
IT Solutions
City of Fort Worth, TX
1000 Throckmorton St
Fort Worth, TX 76102

December 22, 2015

Subject: Letter of Recommendation for David Tschumperle' Regarding G'MIC Software

To whom it may concern:

Recently, David posted a request for feedback from users of G'MIC for use in his evaluation. I am glad to describe my experiences with this software if only to provide some small measure of thanks for what is a remarkable piece of software when nothing has been asked for in return.

My primary work responsibility is Public Safety Dispatch Administrator. We integrate the many systems required to ensure the appropriate resources are dispatched to emergency calls. We have some responsibility for everything from the software running on the Mobile Computers in Police and Fire units, to the call taker and dispatcher and radio system consoles, up to and including server administration.

While my primary responsibilities do not normally include image processing, we end up being a primary point of IT contact for the Police and Fire departments. Once it became known that I was able to provide image processing through GIMP in house without the added expense of a Photoshop license, I became a primary resource when images need to be enhanced.

As such, I was always searching for additions to GIMP to make the job easier. Luckily, I found G'MIC. I can't recall when I first started using G'MIC in GIMP, at least three years. Tasks like removing extraneous objects became trivial with the In-Paint option. Merging photos with seamless blending removed the drudgery from the task.

I am by no means an expert with G'MIC, but the way I edit images now, GIMP is no longer my primary tool, it's the way I get to G'MIC.

Apart from the practical uses of G'MIC, through the tutorials and G'MIC community, I find new, creative, artistic tools that allow me to see what can be, not just what has been set before me. I look forward to each new version of G'MIC, and I can't say that of the other software I use each day.

Sincerely,

Steve Gillow

Bilan et perspectives

Sommaire

9.1 Autres travaux	124
9.2 Projet de recherche	128



Ce chapitre dresse un bilan rapide des travaux effectués, et propose un projet de recherche

9.1 Autres travaux

Les travaux présentés dans les chapitres précédents montrent un panel varié de techniques et d'applications de traitement d'images sur lequel j'ai eu l'occasion d'effectuer mes recherches. Cependant, ils constituent déjà une sélection dans un ensemble plus large d'activités, sélection qui se focalise sur l'utilisation de tenseurs du second ordre pour l'analyse et la modélisation géométrique des images.

Pour donner une vue exhaustive de mes activités, je mentionne ici d'autres travaux réalisés, non détaillés dans ce manuscrit (puisque ne se basant pas sur l'utilisation de modèles tensoriels) :

- **Aide à la colorisation de bandes dessinées** (Période : 2016–2018). Ces travaux, menés en collaboration avec *Sébastien Fourey* (MCF Ensicaen/GREYC), et *David Revoy* (illustrateur français indépendant), nous ont permis de proposer plusieurs algorithmes de segmentation/labelisation pour aider les artistes à coloriser des croquis/dessins en niveaux de gris (*line-arts*) (Fig.9.1 et Fig.9.2). Ces algorithmes, aujourd'hui intégrés dans notre plateforme logicielle *G'MIC*, sont en pratique utilisés par des dessinateurs du monde entier pour accélérer leur travail de colorisation de dessins numériques, travail fastidieux s'il en est. Ce travail a été cité dans plusieurs médias de vulgarisation, dont le magazine *Sciences & Vie Junior* (Fig.9.1), et le rapport d'activités INS2I du CNRS.
 - *Publications associées* : [NC20] (*GRETSI'17*), [R7] (*Conférence internationale et version journal en préparation*).
- **Transfert de couleurs entre images** (Période : 2015–2018). Avec *Julien Rabin* (MCF IUT Caen), nous avons élaboré un algorithme efficace de transfert de couleurs entre deux images, basé sur le recalage variationnel de distributions couleurs RGB utilisant un schéma d'implémentation par flot EDP multi-échelle (Fig.9.3). Cet algorithme est déjà

ACTUS **SCIENCE & TECHNO**

TECHNO

Bulles étanches

Aujourd'hui, mettre une BD en couleur se fait souvent sur ordinateur. Le dessin à l'encre noire est scanné et le coloriste sélectionne avec un stylet chaque zone à coloriser. Hélas, lorsque le trait n'est pas bien fermé – par exemple, si un ballon de foot ne forme pas un cercle parfait –, le programme pédale dans la semoule : la teinte qu'on applique « sort » de l'objet et remplit tout l'arrière-plan de la case. Mais ça, c'était avant qu'une équipe française n'ajoute un algorithme, sorte de recette mathématique, au logiciel de traitement d'image G'MIC. Depuis, l'ordi

comprend beaucoup mieux où il doit s'arrêter : sacré gain de temps pour le coloriste ! Le résultat réussi d'une collaboration entre le CNRS/université de Caen et l'auteur de BD David Revoy. **O.V.**

▲ Dans cette case de David Revoy, le personnage de Pepper est dessiné au trait (1). Le logiciel G'MIC détecte les zones de couleur et leur applique automatiquement une teinte (2), que l'auteur retravaille ensuite en ajoutant effets et détails (3 et 4).

LE CHIFFRE

FIG. 9.1: Aide à la colorisation de bandes dessinées : Ici, notre algorithme génère une colorisation aléatoire de zones segmentées, et l'artiste n'a plus qu'à assigner les bonnes couleurs plutôt que de coloriser tout le dessin "à la main" (extrait de l'article paru dans le magazine *Sciences & Vie Junior*, No 336).

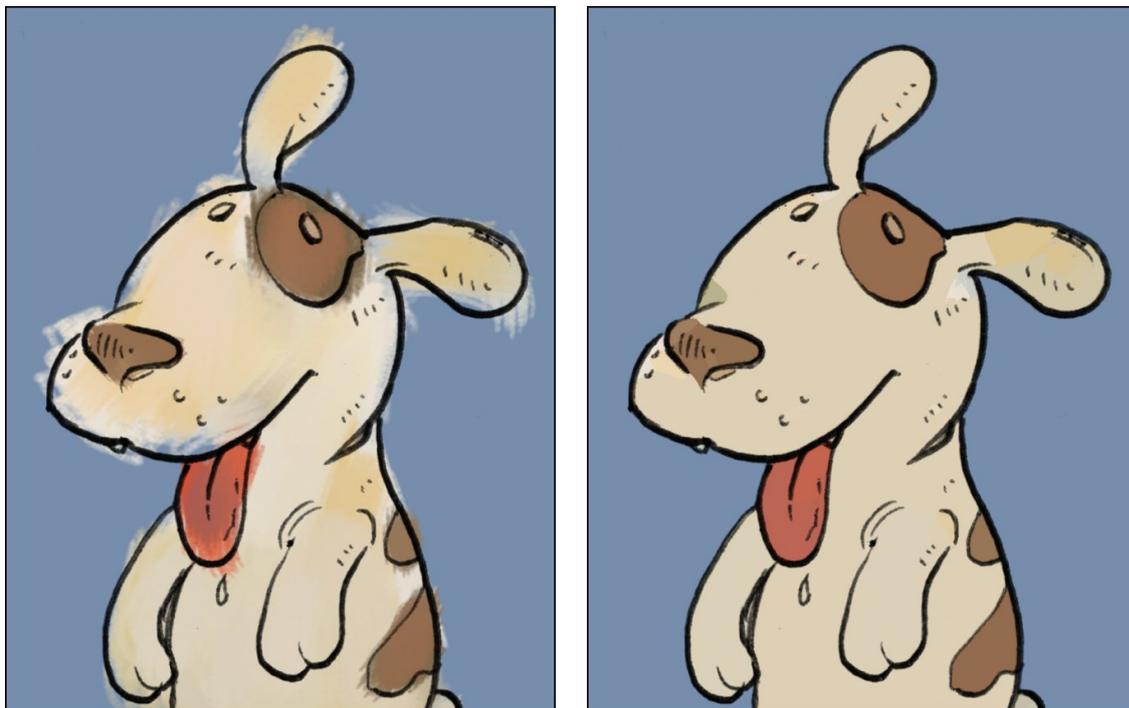


FIG. 9.2: Aide à la colorisation de bandes dessinées : à partir d'une image colorisée grossièrement (image de gauche), notre algorithme est capable de "nettoyer" le calque de couleurs pour coller au mieux à la géométrie du dessin (image de droite).

disponible dans notre plateforme logicielle *G'MIC*.



FIG. 9.3: Transfert de couleurs entre images : Les images sources (images de gauche) sont transformées de manière automatique en leur affectant les couleurs des images modèles (images du centre).

- **Compression de Color LUTs** (Période : 2016). En collaboration avec *Patrick David* (photographe amateur américain), j'ai élaboré un algorithme spécifique de compression (avec perte) de *Color LUTs* (fonctions de transfert de RGB vers RGB). Ces transformations sont de nos jours très appréciées par les photographes, puisqu'elles permettent par exemple de simuler le rendu de vieux films argentiques à partir de photos numériques. Mais en pratique, elles nécessitent le stockage d'images volumineuses couleurs, ce qui peut vite devenir coûteux quand le nombre de transformations augmente (dans notre cas, plus de 350 transformations, disponibles dans notre plateforme *G'MIC*). La compression permet de réduire l'espace utilisé de manière drastique (95% de compression en moyenne), et se base sur une recherche gloutonne de points clés optimaux à stocker, la reconstruction de la fonction globale se réalisant par interpolation par EDP de diffusion avec un schéma multi-échelle.
- **Pseudo-gris pour le rendu d'images scalaires** (Période : 2013). Je me suis également intéressé au problème de la visualisation d'images scalaires dont les valeurs ne sont pas quantifiées de manière importante (typiquement des images scalaires à valeurs flottantes) sur des affichages couleurs, qui eux sont quantifiés (typiquement 256 valeurs par canal couleur). J'ai proposé un algorithme simple de *pseudo-gris* permettant de transformer une image scalaire en image couleur de telle manière que l'affichage de cette image couleur donne une sensation visuelle d'augmentation du nombre de teintes de gris disponible sur l'affichage quantifié.
 - *Publication associée* : [NC15] (*GRETSI'13*).
- **Analyse de flexibles par rayons X** (Période : 2006–2007). En collaboration avec *Jalal Fadili* (Professeur ENSICAEN) et l'*IFP* (Institut Français du Pétrole), nous avons travaillé sur la problématique d'analyse de séquences d'images (rayons X) de flexibles sous-marin faisant passer le pétrole (*risers*). Ces flexibles se déforment sous l'action des courants ma-

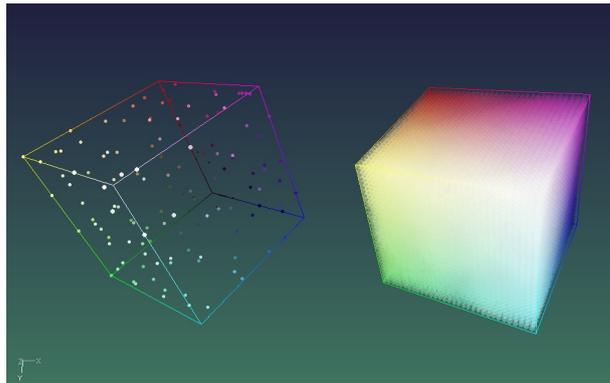


FIG. 9.4: Compression de *Color LUTs* : Une transformation couleur est stockée comme un ensemble de points clés 3D couleurs (gauche), et la reconstruction de la transformation complète se réalise par EDP de diffusion anisotrope dans le volume RGB (droite).

rins et finissent par s'abîmer. Notre but était d'extraire par analyse d'image des suivis de fils d'armure de ces flexibles abîmés pour permettre à l'*IFP* de comparer leurs modèles mathématiques de déformation avec la réalité observée.

- *Publications associées* : [C17] (ICIP'07), [C15] (CVPR'06).

- **Désentrelacement de séquences d'images** (Période : 2004–2005). Ce travail, réalisé en collaboration avec *Bernard Besserer* (MCF, Université de La Rochelle), cherche à réaliser des rendus progressifs de séquences d'images entrelacées, en simulant le rendu de la scène considérée telle qu'il aurait été acquis par une caméra progressive. La prise en compte d'une modélisation fidèle de la physique d'acquisition d'une caméra avec une estimation du mouvement *inter-frames* nous permet de resynthétiser des séquences d'images désentrelacées de bonne qualité (par exemple, avec la possibilité de régénérer du flou de bougé dans le cas de forts mouvements).

- *Publications associées* : [C12] (VIIP'05), [C10] (ICCVG'04).

Bilan : La liste de ces travaux additionnels met en lumière le fait que ma recherche ne se soit pas cantonnée à l'utilisation d'un formalisme mathématique particulier (même si historiquement, les méthodes variationnelles et EDP constituent la base de ma formation), ni à un domaine d'application précis. Au contraire, j'ai été intéressé par l'étude de nombreuses modalités d'images différentes (photographies, vidéos, stéréo, IRMs, dessins, ...), et cette genericité de données caractérise mon parcours de chercheur (et les réalisations logicielles qui vont avec : bibliothèque *CImg* et logiciel *G'MIC*). Du point de vue applicatif, je m'intéresse particulièrement depuis ces dernières années aux problématiques de traitement d'images liées à l'utilisation artistique de l'image (photographie, illustration, cinéma). Cet intérêt grandit en parallèle du nombre d'utilisateurs de mes logiciels, utilisateurs qui sont en grande partie des artistes illustrateurs, photographes ou vidéastes. L'interaction avec cette communauté d'utilisateurs est une source d'inspiration inépuisable pour imaginer et concevoir l'avenir du traitement d'images à des fins artistiques. Ces interactions constituent l'une des originalités et l'un des socles de mon projet de recherche.

9.2 Projet de recherche

9.2.1 Contexte

Ces dernières années, le domaine de la vision par ordinateur a connu une véritable révolution avec l'arrivée des méthodes d'apprentissage profond pour les réseaux de neurones convolutifs (*CNNs*). Ce changement de paradigme a des répercussions à la fois en termes pratiques (efficacité du traitement, performances inégalées) et théoriques. Ces approches, à l'origine de technologies émergentes (telles que les véhicules autonomes), bouleversent de nombreux domaines de la recherche (santé, sécurité, psychologie, . . .), dont le traitement d'images fait partie intégrante.

Certains problèmes considérés comme très difficiles, comme la reconnaissance d'images [SIMONY14, LECUN95, LECUN98]) (Fig.9.7), la colorisation automatique d'images en niveaux de gris [ZHANG16] (Fig.9.5), l'inpainting [IIZUKA17] (Fig.9.5), le transfert de style [GATYS15, JOHNSO16, JING17] (Fig.9.8), ou de couleurs [HE17], ont désormais des solutions acceptables voire quasi parfaites. Faire l'impasse aujourd'hui sur les techniques *CNNs* en traitement d'images n'est donc pas envisageable.



FIG. 9.5: Exemple de colorisation automatique d'images par *CNN*, avec la méthode de [ZHANG16].

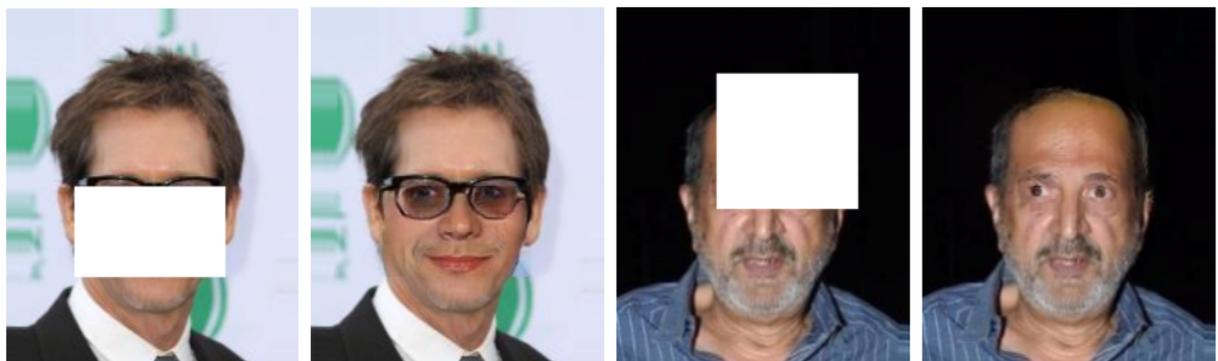


FIG. 9.6: Exemple d'inpainting automatique d'images par *CNN*, avec la méthode de [IIZUKA17].

Ces dernières applications m'intéressent tout particulièrement, puisque elles sont proches d'applications de traitement d'images que j'ai déjà eu l'occasion de traiter précédemment, qu'elles nécessitent des processus d'analyse et des synthèse avancées de la géométrie des images, mais également parce que c'est le type applications très demandées par les artistes utilisateurs de mes logiciels.

9.2.2 Problématique

Cette révolution des *CNNs* apporte également son lot de nouvelles problématiques. En premier lieu, la compréhension des modèles théoriques employés est encore très partielle, à l'image des *GANs* (*Generative Adversarial Networks* [GOODFE14]) qui sont extrêmement populaires pour la génération de données. Les propriétés des algorithmes de l'état de l'art sont le plus souvent inconnues (stabilité, reproductibilité, robustesse, limitations, . . .). Par ailleurs, le gain en performance s'accompagne systématiquement d'un accroissement important de la complexité (temps de calcul hors ligne pour l'apprentissage de plusieurs jours, voire semaines, stockage important des architectures de réseaux profonds comportant des milliards de paramètres). Le problème de la taille croissante des réseaux proposés dans la littérature se pose tout particulièrement. Par exemple, le réseau *VGG-16* [SIMONY14], connu pour être performant en reconnaissance d'images (Fig.9.7), et souvent utilisé comme brique de base pour l'élaboration de réseaux plus larges, nécessite à lui seul déjà plus de *500 Mb* de stockage pour ses coefficients. Cela peut paraître peu, compte tenu du faible coût de la mémoire, mais c'est en réalité un goulot d'étranglement important :

- Le réseau *VGG-16* [SIMONY14] classe des images couleurs d'entrées 224×224 , ce qui à l'heure actuelle est une résolution d'image très modeste. Or, le nombre de coefficients nécessaires pour décrire un tel réseau évolue de manière exponentielle avec l'augmentation de la résolution des images d'entrées. Il est donc aujourd'hui difficile de traiter des images d'entrées de meilleure résolution avec de tels réseaux sans sacrifier des dizaines de giga-octets à cette tâche.
- Ce réseau est souvent utilisé comme un sous-réseau de réseaux plus vastes dans des applications autres que la reconnaissance d'images. La structure des réseaux modernes (parfois plus de 100 couches) implique là encore souvent plusieurs dizaines de giga-octets de stockage nécessaire, pour la plupart des applications.
- Ces tailles importantes freinent clairement la distribution de ces réseaux dans des applications informatiques classiques où le calcul de traitement d'images doit se réaliser en local chez l'utilisateur. Les applications grand-public de traitement d'images utilisant des *CNNs* ont le plus souvent leur cœur de calcul déporté sur des serveurs externes dédiés, avec donc la nécessité pour l'utilisateur d'être connecté pour en profiter. C'est regrettable, parce que la puissance de calcul requise est en général disponible en local chez l'utilisateur !

Ces impératifs d'occupation mémoire sont actuellement peu compatibles avec un usage grand public qui doit permettre de combiner plusieurs filtres paramétrables et personnalisables par un opérateur ou un artiste pour obtenir l'effet désiré.

C'est ainsi que de nombreux utilisateurs (non avertis) nous réclament depuis peu l'intégration de filtres de traitement d'images basées *CNNs* dans notre plateforme logicielle *G'MIC*, alors même qu'il est très improbable que ces mêmes utilisateurs acceptent de sacrifier quelques dizaines de giga-octets de leur disque *pour chaque nouveau filtre ajouté*. Il serait souhaitable de disposer d'algorithmes possédant des performances équivalentes mais utilisant une empreinte mémoire plus réduite, quitte à nécessiter un temps de pré-traitement ou de post-traitement d'images plus conséquent.

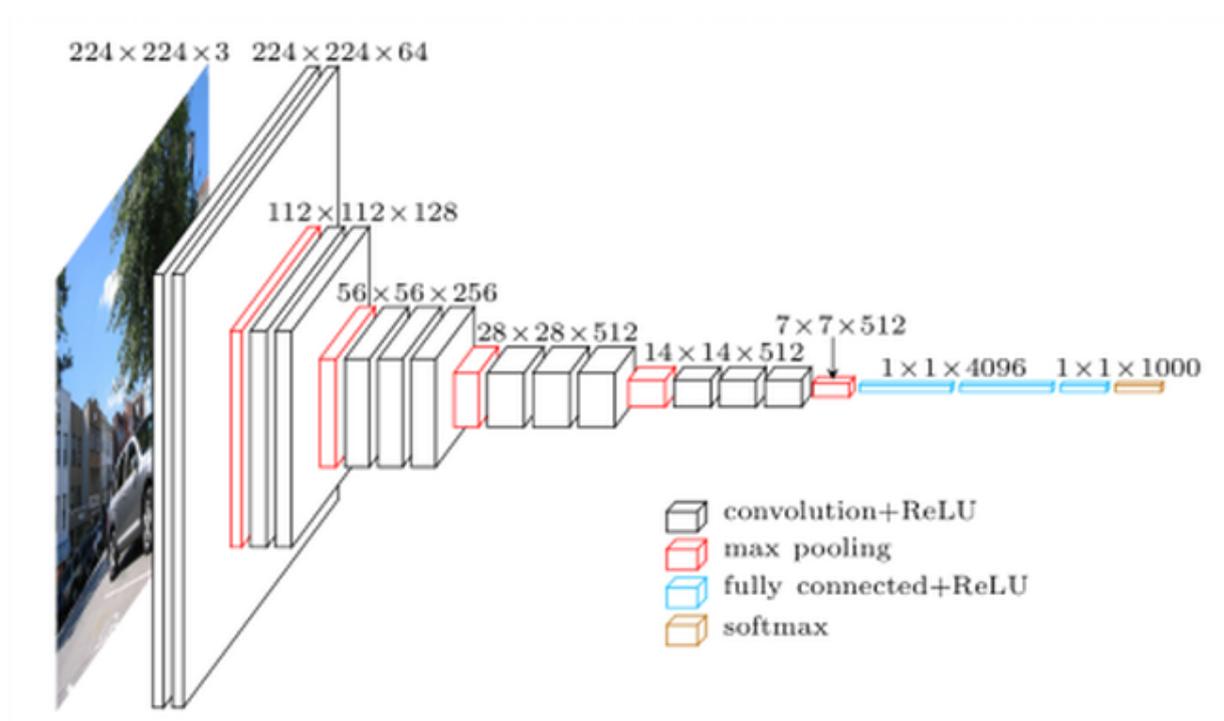


FIG. 9.7: Architecture d'un réseau de neurones convolutionnel simple (VGG-16 [SIMONY14], pour la reconnaissance de contenu d'images). L'ensemble des poids définissant ce réseau nécessite plus de 500 Mb de stockage.

9.2.3 Objectifs

Mon projet de recherche se focalisera donc sur les deux points suivants, qui me semblent les plus importants :

1. **Elaboration et étude de réseaux plus légers.** Un premier objectif de recherche sera donc d'essayer d'approcher les hautes performances des réseaux profonds par des architectures de réseaux plus légères. Dans un très récent travail en synthèse de texture [GALERN17], il a été montré qu'il était possible de définir une telle architecture exploitant des modèles géométriques plus classiques (représentation par dictionnaire de patches) offrant des performances très intéressantes pour un coût en mémoire négligeable. Par ailleurs, d'autres travaux récents [PAPYAN16, SULAM17] ont montré que des liens théoriques existent entre *CNNs*, méthodes basées patches et modélisation parcimonieuse des images, ouvrant des perspectives intéressantes sur la diminution de tailles des architectures *CNNs*. La généralisation de ce type d'approches à d'autres traitements (déconvolution, inpainting, etc) reste néanmoins une gageure. Une attention particulière sera portée aux approches génératives procédurales, très populaires en infographie, qui permettent un traitement en un temps interactif (c'est-à-dire en temps quasi-réel).
2. **Ajout d'attributs géométriques explicites dans les réseaux.** Le second défi réside dans le fait de permettre un meilleur contrôle géométrique par l'utilisateur du résultat produit par le réseau, propriété incontournable en infographie (la plupart des artistes étant opposés aux approches de synthèses d'images totalement automatiques). Ce n'est pas ac-

tuellement pas le cas dans les approches existantes en vision par ordinateur qui visent le plus souvent à optimiser un réseau par un apprentissage entièrement automatique à partir de grandes banques de données. À titre d'exemple, les méthodes de stylisation offrant les résultats les plus impressionnants [JOHNSO16, ULYANO16, GATYS15] (Fig.9.8) mélangent les caractéristiques visuelles de l'image à transformer avec celles d'une image de style ayant les propriétés visuelles souhaitées (Fig.9.8). Hormis le choix d'un exemple de style, l'utilisateur n'a pas véritablement de contrôle sur le résultat produit (conservation de structures comme les yeux d'un visage, suppression d'objets en arrière-plan, etc) ce qui réduit la fonctionnalité de ce type d'outil au fort potentiel. Des travaux récents vont toutefois dans ce sens : la colorisation automatique d'images en niveaux de gris proposée dans [ZHANG17] propose plusieurs palettes de couleurs acceptables afin d'offrir à l'utilisateur des points de contrôle sur le résultat final. Pour résumer, une problématique commune à ces deux défis est d'obtenir une meilleure compréhension des filtres qui sont activés dans les réseaux profonds pour des problématiques de traitement d'images afin d'en augmenter l'efficacité algorithmique et d'en faciliter le contrôle. Arriver à introduire plus d'a-priori géométriques dans les réseaux *CNNs* permettrait également de prendre en compte des heuristiques provenant de connaissances d'experts. Cet objectif va de paire avec le précédent, puisqu'on peut penser que la prise en compte de contraintes géométriques fortes réduire de manière conséquente le nombre de degrés de libertés et par la même, le nombre de coefficients nécessaires pour définir les réseaux.

Pour arriver à ces fins, les techniques suivantes seront privilégiées :

1. **Apprentissage à partir de données parcimonieuses** : les réseaux actuels réalisent pour la plupart leur apprentissage sur des données images "brutes" (tableaux de pixels), et on peut penser qu'alimenter un réseau par des images pré-traitées pour faire ressortir les éléments géométriques saillants des images (contours, textures, etc.) permettrait de réduire le nombre de coefficients nécessaires pour le codage d'un réseau, avec un gain de temps potentiel à la fois en occupation mémoire et en temps d'apprentissage. En revanche, cela impliquerait des étapes de pré-traitements d'images supplémentaires à appliquer avant l'évaluation du réseau.
2. **Ajout d'a priori géométriques dans les fonctions de coût pour l'apprentissage** : on souhaite procéder ici de la même manière qu'avec les termes classiques de régularisation utilisés dans les approches variationnelles en traitement d'images, qui permettent *in-fine* d'imposer des contraintes géométriques fortes (régularité notamment) sur les solutions recherchées. L'apprentissage des *CNNs* est basé sur des algorithmes de descente de gradient, très similaires à ce qui se fait pour la minimisation de fonctionnelles dans les méthodes variationnelles, et on peut imaginer que mieux contraindre géométriquement le processus d'apprentissage permettra, là encore, de diminuer le nombre de coefficients nécessaires à la définition de l'architecture des *CNNs*, en plus d'accélérer le processus d'apprentissage.
3. **Ajout de couches non-linéaires dans les réseaux** : je m'intéresse aussi à la possibilité de créer des architectures de *CNNs* comportant des couches non-convolutionnelles (et non-linéaires). L'observation des masques de convolution appris dans les premières couches d'un réseau (tel que *VGG*, Fig.9.7) montrent en effet que les opérations images réalisées dans ces premières couches se réduisent souvent à l'application de bancs de filtres clas-

siques réalisant du lissage et de la dérivation d'images. Il pourrait être donc souhaitable d'élaborer des couches spécifiques de réseaux dont le rôle serait de réaliser différentes opérations d'analyses géométriques bas-niveau (par exemple, détection de contours, lissage anisotrope, calculs d'attributs géométriques tensoriels), non-linéaires, et avec un nombre de paramètres réduits qui pourraient être appris.

L'idée générale de ce projet de recherche est donc contribuer à remplir le fossé existant entre les méthodes *CNNs* - très performantes mais très coûteuses en occupation mémoire - et les méthodes basées sur une analyse géométrique plus classique (méthodes non-locales basées patchs par exemple), moins performantes, mais très peu coûteuses en mémoire. Les collaborations multiples que je peux avoir avec de nombreux experts du domaine artistique peuvent être riches d'informations géométriques et d'heuristiques pertinentes à intégrer dans les réseaux.

J'illustre ce dernier point avec cette expérience intéressante réalisée en collaboration avec l'artiste illustrateur *David Revoy*, qui était invité au laboratoire GREYC pendant quelques jours. Nous avons comparé les résultats d'un algorithme basé *CNN* de transfert automatique de style (algorithme de [GATYS15]) avec les résultats d'un transfert de style, *peint à la main par l'artiste* (Fig.9.8), ceci, pour une image source et trois images de styles très différentes. Chaque transfert manuel a nécessité entre une et deux heures de travail, l'artiste décrivant les différentes étapes et le raisonnement utilisé pour générer l'image résultat. Ce raisonnement fait très clairement usage de considérations géométriques à la fois locales et non-locales, sur l'analyse des traits de pinceaux (épaisseur de traits, forme des brosses, . . .), des formes et des couleurs (couleurs dominantes, secondaires, . . .). Ces analyses géométriques sont pour la plupart reproductibles avec des algorithmes simples sur un ordinateur, et on peut raisonnablement penser que l'introduction de telles considérations dans les données d'entrées d'un réseau *CNN* (ou dans son processus d'apprentissage) pourrait d'une part améliorer significativement les résultats de transfert automatique, et d'autre part diminuer le nombre de coefficients nécessaires du réseau.

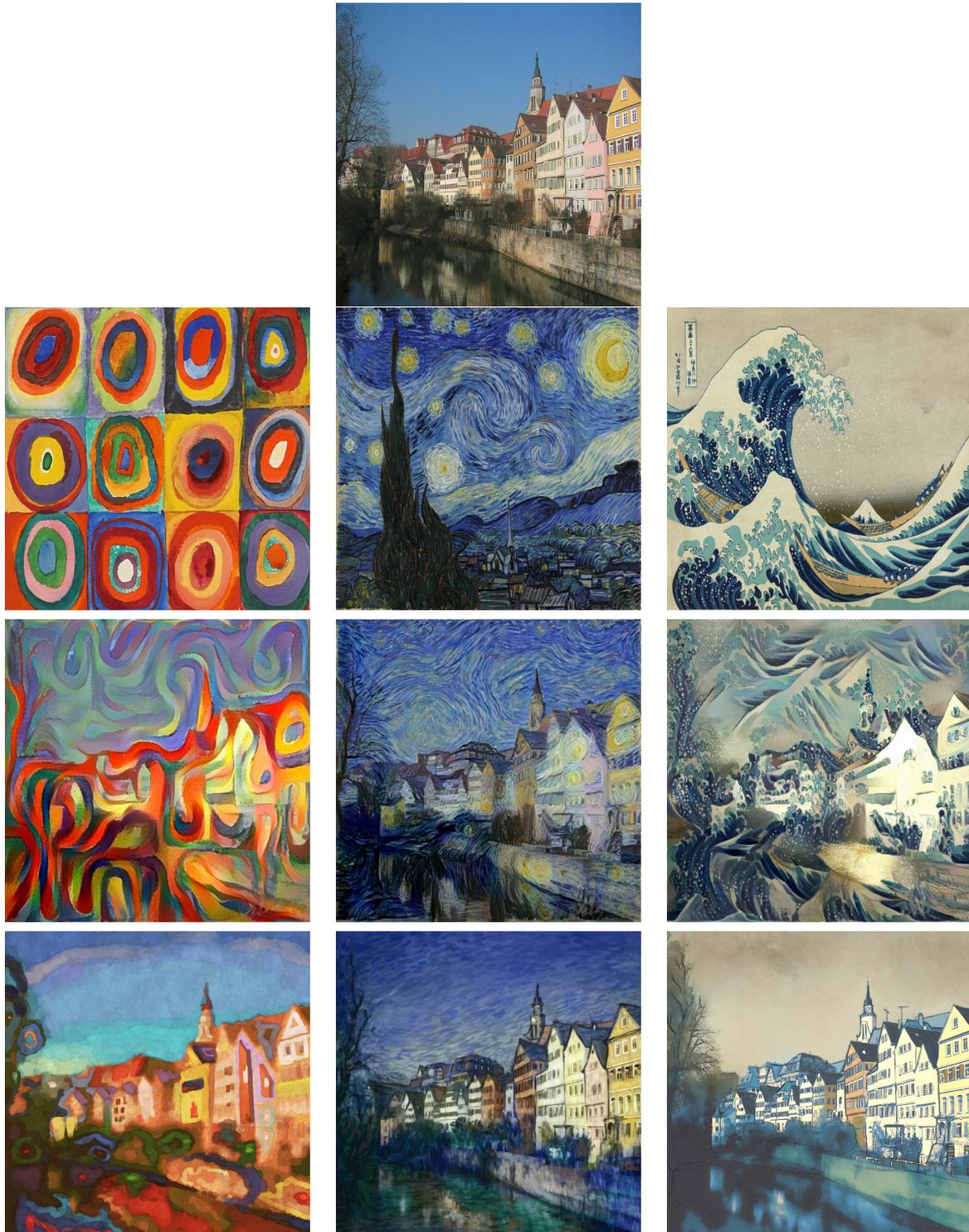


FIG. 9.8: Transfert de style entre deux images : comparaisons de résultats générés automatiquement par un ordinateur, avec des résultats peints manuellement par un artiste.

Première ligne : Image d'entrée à transformer,

Deuxième ligne : Les différentes images de styles que l'on cherche à appliquer,

Troisième ligne : Résultats de transferts de style, réalisés de manière algorithmique, par [GATYS15],

Dernière ligne : Résultats de transferts, peints à la main par l'artiste *David Revoy*.

BIBLIOGRAPHIE

- [ABRAHA12] R. ABRAHAM, J. E. MARSDEN & T. RATIU. « Manifolds, tensor analysis, and applications », vol. 75. Springer Science & Business Media, 2012.
- [ABRAMA82] J.-F. ABRAMATIC & L. M. SILVERMAN. « Nonlinear restoration of noisy images ». *IEEE transactions on pattern analysis and machine intelligence*, 1(2), pages 141–149, 1982.
- [ABRAMO64] M. ABRAMOWITZ & I. A. STEGUN. « Handbook of mathematical functions : with formulas, graphs, and mathematical tables », vol. 55. Courier Corporation, 1964.
- [ADAMS09] A. ADAMS, N. GELFAND, J. DOLSON & M. LEVOY. « Gaussian kd-trees for fast high-dimensional filtering ». Dans *ACM Transactions on Graphics (ToG)*, 28(3), page 21, ACM, 2009.
- [ADOBE] ADOBE. « Content-Aware Fill, Photoshop CS5 ». <http://www.photoshopesentials.com/photo-editing/content-aware-fill-cs5/>.
- [AGARWA04] A. AGARWALA, M. DONTCHEVA, M. AGRAWALA, S. DRUCKER, A. COLBURN, B. CURLESS, D. SALESIN & M. COHEN. « Interactive digital photomontage ». Dans *ACM Transactions on Graphics*, 23(3), pages 294–302, ACM, 2004.
- [ALEXAN02] D. ALEXANDER, G. BARKER & S. ARRIDGE. « Detection and modeling of non-Gaussian apparent diffusion coefficient profiles in human brain data ». *Magnetic Resonance Imaging*, vol. 48, pages 331–340, 2002.
- [ALEXAN07] D. C. ALEXANDER. « Axon radius measurements in vivo from diffusion MRI : a feasibility study ». Dans *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, IEEE, 2007.
- [ALVARE92] L. ALVAREZ, P.-L. LIONS & J.-M. MOREL. « Image selective smoothing and edge detection by nonlinear diffusion. II ». *SIAM Journal on numerical analysis*, 29(3), pages 845–866, 1992.
- [ALVARE93] L. ALVAREZ, F. GUICHARD, P.-L. LIONS & J.-M. MOREL. « Axioms and fundamental equations of image processing ». *Archive for rational mechanics and analysis*, 123(3), pages 199–257, 1993.
- [ALVARE94] L. ALVAREZ & L. MAZORRA. « Signal and image restoration using shock filters and anisotropic diffusion ». *SIAM Journal on Numerical Analysis*, 31(2), pages 590–605, 1994.
- [ALVARE00] L. ALVAREZ, R. DERICHE & F. SANTANA. « Recursivity and PDEs in image processing ». Dans *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, vol. 1, pages 242–248, IEEE, 2000.
- [ANDREW99] G. ANDREWS, R. ASKEY & R. ROY. « Special Functions ». Cambridge University Press, 1999.
- [ARIAS11] P. ARIAS, G. FACCILOLO, V. CASELLES & G. SAPIRO. « A Variational Framework for Exemplar-Based Image Inpainting ». *Int. J. Comput. Vision*, 93(3), pages 319–347, juillet 2011. ISSN 0920-5691.

- [ARIAS12] P. ARIAS, V. CASELLES & G. FACCIOLO. « Analysis of a variational framework for exemplar-based image inpainting ». *Multiscale Modeling & Simulation*, 10(2), pages 473–514, 2012.
- [ASHIKH01] M. ASHIKHMIN. « Synthesizing natural textures ». Dans *Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 217–226, ACM, 2001.
- [ASHIKH03] N. ASHIKHMIN. « Fast texture transfer ». *IEEE Computer Graphics and Applications*, 23(4), pages 38–43, 2003.
- [ASSAF98] Y. ASSAF & Y. COHEN. « In vivo and in vitro bi-exponential diffusion of N-acetyl aspartate (NAA) in rat brain : a potential structural probe ? ». *NMR Biomed*, vol. 11, pages 67–74, 1998.
- [ASSAF00] Y. ASSAF & Y. COHEN. « Assignment of the Water Slow-Diffusing Component in the Central Nervous System Using q-Space DiffusionMRS : Implications for Fiber Tract Imaging ». *Magn. Res. Med.*, vol. 43, pages 191–199, 2000.
- [ASSAF05] Y. ASSAF & P. J. BASSER. « Composite hindered and restricted model of diffusion (CHARMED) MR imaging of the human brain ». *NeuroImage*, vol. 27, pages 48–58, 2005.
- [ASSEML07] H.-E. ASSEMLAL, D. TSCHUMPERLÉ & L. BRUN. « Fiber tracking on hardi data using robust odF fields ». Dans *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 3, pages III–133, IEEE, 2007.
- [ASSEML08] H. ASSEMLAL, D. TSCHUMPERLÉ & L. BRUN. « Efficient Computation of PDF-based Characteristics from Diffusion MR Signal. ». Dans *MICCAI*, pages 70–78, New York/USA, September 2008.
- [ASSEML10A] H.-E. ASSEMLAL. « Diffusion MR image analysis for the estimation of tissues local architecture ». Theses, Université de Caen, janvier 2010.
- [ASSEML10B] H.-E. ASSEMLAL. « Traitement et analyse d’images IRM de diffusion pour l’estimation de l’architecture locale des tissus ». Theses, Université de Caen, janvier 2010.
- [AUBERT02] G. AUBERT & P. KORNPBST. « Mathematical Problems in Image Processing : PDEs and the Calculus of Variations, vol. 147 of App. Mathem ». Springer-Verlag, 2002.
- [BAKER04] S. BAKER & I. MATTHEWS. « Lucas-kanade 20 years on : A unifying framework ». *International journal of computer vision*, 56(3), pages 221–255, 2004.
- [BALLE01] C. BALLESTER, M. BERTALMIO, V. CASELLES, G. SAPIRO & J. VERDERA. « Filling-in by joint interpolation of vector fields and gray levels ». *IEEE Trans. on Image Process.*, 10(8), pages 1200–1211, 2001.
- [BARASH01] D. BARASH. « Bilateral filtering and anisotropic diffusion : Towards a unified viewpoint ». Dans *International Conference on Scale-Space Theories in Computer Vision*, pages 273–280, Springer, 2001.
- [BARASH02] D. BARASH. « Fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation ». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(6), pages 844–847, 2002.
- [BARASH04] D. BARASH & D. COMANICIU. « A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift ». *Image and Vision Computing*, 22(1), pages 73–81, 2004.
- [BARNES09] C. BARNES, E. SHECHTMAN, A. FINKELSTEIN & D. B. GOLDMAN. « PatchMatch : A Randomized Correspondence Algorithm for Structural Image Editing ». *ACM T. Graphic.*, 28(3), août 2009.
- [BARNES10] C. BARNES, E. SHECHTMAN, D. B. GOLDMAN & A. FINKELSTEIN. « The Generalized PatchMatch Correspondence Algorithm ». Dans *ECCV*, septembre 2010.

- [BARTM.94] BART M. TER HAAR ROMENY. « Geometry-driven diffusion in computer vision ». *Computational imaging and vision*, Kluwer Academic Publishers, vol. 1, 1994.
- [BASSER94] P. J. BASSER, J. MATTIELLO & D. LEBIHAN. « Estimation of the effective self-diffusion tensor from the NMR spin echo ». *Journal of Magnetic Resonance*, vol. 103, pages 247–254, 1994.
- [BASSER98] P. BASSER & C. PIERPAOLI. « A Simplified Method to Measure the Diffusion Tensor from Seven MR Images ». *Magn. Res. Med.*, vol. 39, pages 928–934, 1998.
- [BASSER00] P. BASSER, S. PAJEVIC, C. PIERPAOLI, J. DUDA & A. ALDROUBI. « In vivo fiber tractography using DT-MRI data ». *Magn. Res. Med.*, 44(4), pages 625–632, 2000.
- [BASSER02] P. BASSER. « Relationships Between Diffusion Tensor and q-Space MRI ». *Magn. Res. Med.*, 47(2), pages 392–397, 2002.
- [BASU06] S. BASU, T. FLETCHER & R. WHITAKER. « Rician noise removal in diffusion tensor MRI ». Dans *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 117–125, Springer, 2006.
- [BECKER00] J. BECKER, T. PREUSSER & M. RUMPF. « PDE methods in flow simulation post processing ». *Computing and Visualization in Science*, 3(3), pages 159–167, 2000.
- [BENTLE75] J. L. BENTLEY. « Multidimensional binary search trees used for associative searching ». *Communications of the ACM*, 18(9), pages 509–517, 1975.
- [BERNST89] M. BERNSTEIN, D. THOMASSON & W. PERMAN. « Improved detectability in low signal-to-noise ratio magnetic resonance images by means of a phase-corrected real reconstruction ». *Medical Physics*, vol. 16, page 813, 1989.
- [BERTAL00] M. BERTALMIO, G. SAPIRO, V. CASELLES & C. BALLESTER. « Image inpainting ». Dans *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424, ACM Press/Addison-Wesley Publishing Co., 2000.
- [BERTAL01A] M. BERTALMIO, A. L. BERTOZZI & G. SAPIRO. « Navier-stokes, fluid dynamics, and image and video inpainting ». Dans *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, pages I–355, 2001.
- [BERTAL01B] M. BERTALMIO, L.-T. CHENG, S. OSHER & G. SAPIRO. « Variational problems and partial differential equations on implicit surfaces ». *Journal of Computational Physics*, 174(2), pages 759–780, 2001.
- [BERTAL03] M. BERTALMIO, L. VESE, G. SAPIRO & S. OSHER. « Simultaneous structure and texture image inpainting ». *IEEE transactions on image processing*, 12(8), pages 882–889, 2003.
- [BIEDEN81] L. C. BIEDENHARN & J. D. LOUCK. « Angular momentum in quantum physics ». Addison-Wesley Publishing Co., Reading, Mass., 1981. ISBN 0-201-13508-6.
- [BLACK98] M. J. BLACK, G. SAPIRO, D. H. MARIMONT & D. HEEGER. « Robust anisotropic diffusion ». *IEEE Transactions on image processing*, 7(3), pages 421–432, 1998.
- [BLANC-95] L. BLANC-FÉRAUD, P. CHARBONNIER, G. AUBERT & M. BARLAUD. « Nonlinear image processing : modeling and fast algorithm for regularization with edge detection ». Dans *Image Processing, 1995. Proceedings., International Conference on*, vol. 1, pages 474–477, IEEE, 1995.
- [BLANCO97] M. A. BLANCO, M. FLÓREZ & M. BERMEJO. « Evaluation of the rotation matrices in the basis of real spherical harmonics ». *J. Mol. Struct.*, vol. 419, pages 19–27, 1997.
- [BLOMGR98] P. BLOMGREN & T. F. CHAN. « Color TV : Total variation methods for restoration of vector-valued images ». *IEEE transactions on image processing*, 7(3), pages 304–309, 1998.

- [BORNAR02] R. BORNARD, E. LECAN, L. LABORELLI & J.-H. CHENOT. « Missing data correction in still images and image sequences ». Dans *Proceedings of the tenth ACM international conference on Multimedia*, pages 355–361, 2002.
- [BORNEM07] F. BORNEMANN & T. MÄRZ. « Fast image inpainting based on coherence transport ». *J. Math. Imaging Vis.*, 28(3), pages 259–278, 2007.
- [BOUHAM07] A. BOUHAMIDI & K. JBILOU. « Sylvester Tikhonov-regularization methods in image restoration ». *Journal of Computational and Applied Mathematics*, 206(1), pages 86–98, 2007.
- [BOULAN07] J. BOULANGER, C. KERVRANN & P. BOUTHEMY. « Space-time adaptation for patch-based image sequence restoration ». *IEEE transactions on pattern analysis and machine intelligence*, 29(6), 2007.
- [BREIMA68] L. BREIMAN. « Probability Addison-Wesley ». *Reading, Mass*, 1968.
- [BROOKS07] S. BROOKS. « Mixed Media Painting and Portraiture ». *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, pages 1041–1054, 2007. ISSN 1077-2626.
- [BROX07] T. BROX & D. CREMERS. « Iterated nonlocal means for texture restoration ». Dans *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 13–24, Springer, 2007.
- [BRUCKN07] S. BRUCKNER & M. E. GRÖLLER. « Style transfer functions for illustrative volume rendering ». Dans *Computer Graphics Forum*, 26(3), pages 715–724, Wiley Online Library, 2007.
- [BUADES05A] A. BUADES, B. COLL & J.-M. MOREL. « A non-local algorithm for image denoising ». Dans *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, pages 60–65, IEEE, 2005.
- [BUADES05B] A. BUADES, B. COLL & J.-M. MOREL. « A review of image denoising algorithms, with a new one ». *Multiscale Model. Sim.*, 4(2), pages 490–530, 2005.
- [BUYSS15A] P. BUYSENS, M. DAISY, D. TSCHUMPERLÉ & O. LÉZORAY. « Depth-aware patch-based image disocclusion for virtual view synthesis ». Dans *SIGGRAPH Asia 2015 Technical Briefs*, page 2, 2015.
- [BUYSS15B] P. BUYSENS, M. DAISY, D. TSCHUMPERLÉ & O. LÉZORAY. « Exemplar-based Inpainting : Technical Review and new Heuristics for better Geometric Reconstructions ». *Image Processing, IEEE Transactions on*, 24(6), pages 1809–1824, 2015.
- [BUYSS15C] P. BUYSENS, M. DAISY, D. TSCHUMPERLÉ & O. LÉZORAY. « Superpixel-based depth map inpainting for RGB-D view synthesis ». Dans *Image Processing (ICIP), 2015 22th IEEE International Conference on*, Québec City, Canada, septembre 2015.
- [CABRAL93] B. CABRAL & L. C. LEEDOM. « Imaging vector fields using line integral convolution ». Dans *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 263–270, ACM, 1993.
- [CAI08] J.-F. CAI, R. H. CHAN & Z. SHEN. « A framelet-based image inpainting algorithm ». *Appl. Comput. Harmon. A.*, 24(2), pages 131–149, 2008.
- [CALLAG91A] P. CALLAGHAN. « Principles of Nuclear Magnetic Resonance Microscopy ». Oxford University Press, USA, 1991.
- [CALLAG91B] P. CALLAGHAN, A. COY, D. MACGOWAN, K. PACKER & F. ZELAYA. « Diffraction-like effects in NMR diffusion studies of fluids in porous solids ». *Nature*, 351(6326), pages 467–469, 1991.
- [CALLEN00] A. CALLEN. « The art of Impressionism : painting technique & the making of modernity ». Yale University Press, 2000.

- [CAO11] F. CAO, Y. GOUSSEAU, S. MASNOU & P. PÉREZ. « Geometrically Guided Exemplar-Based Inpainting ». *SIAM J. Imag. Sciences*, 4(4), pages 1143–1179, 2011.
- [CARMON98] R. A. CARMONA & S. ZHONG. « Adaptive smoothing respecting feature directions ». *IEEE Transactions on Image Processing*, 7(3), pages 353–358, 1998.
- [CERCIG06] M. CERCIGNANI & D. ALEXANDER. « Optimal acquisition schemes for in vivo quantitative magnetization transfer MRI ». *Magn. Res. Med.*, vol. 56, pages 803–810, 2006.
- [CHAMBO97] A. CHAMBOLLE & P.-L. LIONS. « Image recovery via total variation minimization and related problems ». *Numerische Mathematik*, 76(2), pages 167–188, 1997.
- [CHAN01] T. F. CHAN & J. SHEN. « Nontexture inpainting by curvature-driven diffusions ». *Journal of Visual Communication and Image Representation*, 12(4), pages 436–449, 2001.
- [CHAN06] T. F. CHAN, J. SHEN & H.-M. ZHOU. « Total variation wavelet inpainting ». *J. Math. Imaging Vis.*, 25(1), pages 107–125, 2006.
- [CHARBO94] P. CHARBONNIER, L. BLANC-FERAUD, G. AUBERT & M. BARLAUD. « Two deterministic half-quadratic regularization algorithms for computed imaging ». Dans *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, vol. 2, pages 168–172, IEEE, 1994.
- [CHEFD'04] C. CHEFD'HOTEL, D. TSCHUMPERLÉ, R. DERICHE & O. FAUGERAS. « Regularizing flows for constrained matrix-valued images ». *Journal of Mathematical Imaging and Vision*, 20(1), pages 147–162, 2004.
- [CHEN04A] H. CHEN, Z. LIU, C. ROSE, Y. XU, H.-Y. SHUM & D. SALESIN. « Example-based composite sketching of human portraits ». Dans *NPAP '04 : Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*, pages 95–153, 2004. ISBN 1-58113-887-3.
- [CHEN04B] Y. CHEN, W. GUO, Q. ZENG, X. YAN, F. HUANG, H. ZHANG, G. HE, B. VEMURI & Y. LIU. « Estimation, smoothing, and characterization of apparent diffusion coefficient profiles from high angular resolution dwi ». *CVPR*, vol. 1, pages 588–593, 2004.
- [CHEN07] J. CHEN, S. PARIS & F. DURAND. « Real-time edge-aware image processing with the bilateral grid ». *ACM Transactions on Graphics (TOG)*, 26(3), page 103, 2007.
- [CHENG14] F.-H. CHENG & M.-D. LOO. « An Image Inpainting Method for Stereoscopic Images Based on Filling Route ». Dans *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2014 Tenth International Conference on*, pages 399–402, 2014.
- [CLARK00] C. CLARK & D. LE BIHAN. « Water diffusion and anisotropy at high b values in the human brain ». *Magn. Res. Med.*, vol. 44, pages 852–859, 2000.
- [COHEN02] Y. COHEN & Y. ASSAF. « High b-value q-space analyzed diffusion-weighted MRS and MRI in neuronal tissues - a technical review ». *NMR Biomed*, vol. 15, pages 516–542, 2002.
- [COLLOM05] J. P. COLLOMOSSE, D. ROWNTREE & P. M. HALL. « Stroke Surfaces : Temporally Coherent Artistic Animations from Video ». *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, pages 540–549, 2005. ISSN 1077-2626.
- [COOK07] P. COOK, M. SYMMS, P. BOULBY & D. ALEXANDER. « Optimal acquisition orders of diffusion-weighted MRI measurements ». *Journal of Magnetic Resonance Imaging*, 25(5), pages 1051–8, 2007.
- [CORY90] D. CORY & A. GARROWAY. « Measurement of translational displacement probabilities by NMR : an indicator of compartmentation ». *Magn. Res. Med.*, 14(3), pages 435–444, 1990.

- [CRIMIN04] A. CRIMINISI, P. PÉREZ & K. TOYAMA. « Region filling and object removal by exemplar-based image inpainting ». *IEEE Transactions on image processing*, 13(9), pages 1200–1212, 2004.
- [CROSS83] G. R. CROSS & A. K. JAIN. « Markov random field texture models ». *IEEE Trans. Pattern Anal. Mach. Intell.*, 1(1), pages 25–39, 1983.
- [CROW84] F. C. CROW. « Summed-area tables for texture mapping ». *ACM SIGGRAPH computer graphics*, 18(3), pages 207–212, 1984.
- [CURTIS97] C. J. CURTIS, S. E. ANDERSON, J. E. SEIMS, K. W. FLEISCHER & D. H. SALESIN. « Computer-generated watercolor ». Dans *SIGGRAPH '97 : Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 421–430, 1997. ISBN 0-89791-896-7.
- [DABOV07] K. DABOV, A. FOI, V. KATKOVNIK & K. EGIAZARIAN. « Image denoising by sparse 3-D transform-domain collaborative filtering ». *IEEE Transactions on image processing*, 16(8), pages 2080–2095, 2007.
- [DAISY13A] M. DAISY, D. TSCHUMPERLÉ & O. LÉZORAY. « A Fast Spatial Patch Blending Algorithm for Artefact Reduction in Pattern-based Image Inpainting ». Dans *SIGGRAPH Asia 2013 Technical Briefs*, pages 8 :1–8 :4, ACM, 2013.
- [DAISY13B] M. DAISY, D. TSCHUMPERLÉ & O. LÉZORAY. « Spatial patch blending for artefact reduction in pattern-based inpainting techniques ». Dans *CAIP*, pages 523–530, Springer, 2013.
- [DAISY14] M. DAISY, P. BUYSENS, D. TSCHUMPERLÉ & O. LÉZORAY. « A smarter exemplar-based inpainting algorithm using local and global heuristics for more geometric coherence ». Dans *ICIP*, pages 1–5, IEEE, 2014.
- [DAISY15] M. DAISY. « Inpainting basé motif d'images et de vidéos appliqué aux données stéréoscopiques avec carte de profondeur ». Theses, Université de Caen Normandie, décembre 2015.
- [DARABI12] S. DARABI, E. SHECHTMAN, C. BARNES, D. B. GOLDMAN & P. SEN. « Image melding : combining inconsistent images using patch-based synthesis ». *ACM T. Graphic.*, 31(4), page 82, 2012.
- [DARIBO10] I. DARIBO & B. PESQUET-POPESCU. « Depth-aided image inpainting for novel view synthesis ». Dans *IEEE International Workshop on Multimedia Signal Processing*, pages 167–170, 2010.
- [DECARL02] D. DECARLO & A. SANTELLA. « Stylization and abstraction of photographs ». Dans *SIGGRAPH*, pages 769–776, 2002.
- [DECARL03] D. DECARLO, A. FINKELSTEIN, S. RUSINKIEWICZ & A. SANTELLA. « Suggestive contours for conveying shape ». *ACM Trans. Graph.*, 22(3), pages 848–855, 2003. ISSN 0730-0301.
- [DECAUD96] P. DECAUDIN. « Abstract Cartoon-Looking Rendering of 3D-Scenes », 1996.
- [DEMANE03] L. DEMANET, B. SONG & T. CHAN. « Image inpainting by correspondence maps : a deterministic approach ». *Appl. Comput. Math.*, vol. 1100, pages 217–50, 2003.
- [DERICH95] R. DERICHE & O. FAUGERAS. « Les EDP en traitement des images et vision par ordinateur ». Thèse de doctorat, INRIA, 1995.
- [DESCOT06] M. DESCOTEAUX, E. ANGELINO, S. FITZGIBBONS & R. DERICHE. « Apparent diffusion coefficients from high angular resolution diffusion imaging : estimation and applications. ». *Magn Reson Med*, 56(2), pages 395–410, 2006.

- [DESCOT07] M. DESCOTEAUX, E. ANGELINO, S. FITZGIBBONS & R. DERICHE. « Regularized, Fast and Robust Analytical Q-Ball Imaging ». *Magn. Res. Med.*, vol. 58, pages 497–510, 2007.
- [DESCOT08] M. DESCOTEAUX, N. WIEST-DAESSLE, S. PRIMA, C. BARILLOT & R. DERICHE. « Impact of Rician Adapted Non-Local Means Filtering on HARDI ». Dans *MICCAI*, pages 122–130, New York/USA, September 2008.
- [DEUSSE00A] O. DEUSSEN, S. HILLER, C. VAN OVERVELD & T. STROTHOTTE. « Floating Points : A Method for Computing Stipple Drawings ». *Computer Graphics Forum*, vol. 19, pages 40–51, 2000.
- [DEUSSE00B] O. DEUSSEN & T. STROTHOTTE. « Computer-generated pen-and-ink illustration of trees ». Dans *SIGGRAPH*, pages 13–18, 2000. ISBN 1-58113-208-5.
- [DI ZEN86] S. DI ZENZO. « A note on the gradient of a multi-image ». *Computer vision, graphics, and image processing*, 33(1), pages 116–125, 1986.
- [D.K. J99] A. S. D.K. JONES. « Optimal strategies for measuring diffusion in anisotropic systems by magnetic resonance imaging ». *Magn. Res. Med.*, 42(3), pages 515–525, 1999.
- [DRORI03] I. DRORI, D. COHEN-OR & H. YESHURUN. « Fragment-based image completion ». Dans *ACM T. Graphic.*, 22(3), pages 303–312, ACM, 2003.
- [D.TSCH08] D. TSCHUMPERLÉ. « G'MIC, a full-featured open-source framework for image processing ». <http://gmic.eu>, 2008.
- [DURAND01] F. DURAND, V. OSTROMOUKHOV, M. MILLER, F. DURANLEAU & J. DORSEY. « Decoupling Strokes and High-Level Attributes for Interactive Traditional Drawing ». Dans *Eurographics Workshop on Rendering Techniques*, pages 71–82, 2001.
- [EDELST84] W. EDELSTEIN, P. BOTTOMLEY & L. PFEIFER. « A signal-to-noise calibration procedure for NMR imaging systems ». *Medical Physics*, vol. 11, page 180, 1984.
- [EFROS99] A. A. EFROS & T. K. LEUNG. « Texture synthesis by non-parametric sampling ». Dans *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, pages 1033–1038, IEEE, 1999.
- [ELAD05] M. ELAD, J.-L. STARCK, P. QUERRE & D. L. DONOHO. « Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA) ». *Applied and Computational Harmonic Analysis*, 19(3), pages 340–358, 2005.
- [ELAD06] M. ELAD & M. AHARON. « Image denoising via sparse and redundant representations over learned dictionaries ». *IEEE T. Image Process.*, 15(12), pages 3736–3745, 2006.
- [ELDER01] J. H. ELDER & R. M. GOLDBERG. « Image editing in the contour domain ». *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(3), pages 291–296, 2001.
- [FADILI07] M.-J. FADILI, J.-L. STARCK & L. BOUBCHIR. « Morphological diversity and sparse image denoising ». Dans *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 1, pages I-589, IEEE, 2007.
- [FAUGER04] O. FAUGERAS, G. ADDE, G. CHARPIAT, C. CHEFD'HOTEL, M. CLERC, T. DENEUX, R. DERICHE, G. HERMOSILLO, R. KERIVEN, P. KORNPBST, J. KYBIC, C. LENGLET, L. LOPEZ-PEREZ, T. PAPADOPOULOU, J.-P. PONS, F. SEGONNE, B. THIRION, D. TSCHUMPERLÉ, T. VIÉVILLE & N. WOTAWA. « Variational, geometric, and statistical methods for modeling brain anatomy and function ». *NeuroImage*, vol. 23, pages S46–S55, 2004.
- [FEHN04] C. FEHN. « Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV ». Dans *Electronic Imaging 2004*, pages 93–104, International Society for Optics and Photonics, 2004.

- [FRANK02] L. FRANK. « Characterization of anisotropy in high angular resolution diffusion-weighted mri ». *Magn. Res. Med.*, vol. 47, pages 1083–1099, 2002.
- [G. REG10] D. G. REGAN & P. KUCHEL. « NMR Studies of Diffusion-Coherence Phenomena in Red Cell Suspensions : Current Status ». vol. 43, pages 45–54, 03 2010.
- [G.09] G. G. « Digital Retouching for Fashion Beauty & Portrait Photography ». Eureka Imaging Publications, 2009.
- [GALERN17] B. GALERNE, A. LECLAIRE & J. RABIN. « Semi-Discrete Optimal Transport in Patch Space for Enriching Gaussian Textures ». Dans *International Conference on Geometric Science of Information*, pages 100–108, Springer, 2017.
- [GATYS15] L. A. GATYS, A. S. ECKER & M. BETHGE. « A neural algorithm of artistic style ». *arXiv preprint arXiv :1508.06576*, 2015.
- [GAUTIE11] J. GAUTIER, O. LE MEUR & C. GUILLEMOT. « Depth-based image completion for view synthesis ». Dans *3DTV Conference : The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4, 2011.
- [GEMAN85] S. GEMAN & D. MCCLURE. « Bayesian image analysis : An application to single photon emission tomography ». *Amer. Statist. Assoc*, pages 12–18, 1985.
- [GIMP08] GIMP. « Image manipulation program ». <http://www.gimp.org>, 2008.
- [GOODFE14] I. GOODFELLOW, J. POUGET-ABADIE, M. MIRZA, B. XU, D. WARDE-FARLEY, S. OZAIR, A. COURVILLE & Y. BENGIO. « Generative adversarial nets ». Dans *Advances in neural information processing systems*, pages 2672–2680, 2014.
- [GRANAD12] M. GRANADOS, J. TOMPKIN, K. KIM, O. GRAU, J. KAUTZ & C. THEOBALT. « How Not to Be Seen-Object Removal from Videos of Crowded Scenes ». Dans *Computer Graphics Forum*, 31(2pt1), pages 219–228, Wiley Online Library, 2012.
- [GRANLU95] G. GRANLUND & H. KNUTSSON. « Signal Processing for Computer Vision ». Kluwer Academic Publish., 1995.
- [GREEN90] P. J. GREEN. « Bayesian reconstructions from emission tomography data using a modified EM algorithm ». *IEEE transactions on medical imaging*, 9(1), pages 84–93, 1990.
- [GUDBJA95] H. GUDBJARTSSON & S. PATZ. « The Rician distribution of noisy MRI data ». *Magn. Res. Med.*, vol. 34, pages 910–914, 1995.
- [GUILLE13] C. GUILLEMOT, M. TURKAN, O. LE MEUR, M. EBDELLI ET AL. « Object removal and loss concealment using neighbor embedding ». *Eurasip Journal on Signal Processing : Image Communication*, 2013.
- [GUILLE14] C. GUILLEMOT & O. LE MEUR. « Image inpainting : Overview and recent advances ». *Signal Processing Magazine, IEEE*, 31(1), pages 127–144, 2014.
- [HANDAM23] J. HANDAMARD. « Lectures on the Cauchy Problems in Linear Partial Differential Equations,(1923) », 1923.
- [HAWORT01] C. HAWORTH, A. M. PEACOCK & D. RENSHAW. « Performance of reference block updating techniques when tracking with the block matching algorithm ». Dans *Image Processing, 2001. Proceedings. 2001 International Conference on*, vol. 1, pages 365–368, IEEE, 2001.
- [HAYS07] J. HAYS & A. A. EFROS. « Scene completion using millions of photographs ». Dans *ACM Transactions on Graphics (TOG)*, 26(3), page 4, ACM, 2007.
- [HE12A] K. HE & J. SUN. « Statistics of Patch Offsets for Image Completion ». Dans *European Conference on Computer Vision*, pages 16–29, Springer-Verlag, 2012. ISBN 978-3-642-33708-6.

- [HE12B] K. HE & J. SUN. « Computing nearest-neighbor fields via propagation-assisted kd-trees ». Dans *CVPR*, pages 111–118, IEEE, 2012.
- [HE17] M. HE, J. LIAO, L. YUAN & P. V. SANDER. « Neural Color Transfer between Images ». *arXiv preprint arXiv :1710.00756*, 2017.
- [HEALY98] D. HEALY, H. HENDRIKS & P. KIM. « Spherical Deconvolution ». *Journal of Multivariate Analysis*, 67(1), pages 1–22, 1998.
- [HENKEL85] R. HENKELMAN. « Measurement of signal intensities in the presence of noise in MR images ». *Med. Phys.*, vol. 12, pages 232–233, 1985.
- [HENKEL87] R. HENKELMAN & M. BRONSKILL. « Artifacts in magnetic resonance imaging ». Pergamon Press, 1987.
- [HERTZM98] A. HERTZMANN. « Painterly rendering with curved brush strokes of multiple sizes ». Dans *SIGGRAPH*, pages 453–460, 1998.
- [HERTZM00] A. HERTZMANN & D. ZORIN. « Illustrating smooth surfaces ». Dans *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 517–526, ACM Press/Addison-Wesley Publishing Co., 2000.
- [HERTZM01A] A. HERTZMANN. « Paint By Relaxation ». Dans *CGI '01 : Computer Graphics International 2001*, pages 47–54, 2001. ISBN 0-7695-1007-8.
- [HERTZM01B] A. HERTZMANN, C. E. JACOBS, N. OLIVER, B. CURLESS & D. H. SALESIN. « Image analogies ». Dans *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340, ACM, 2001.
- [HERTZM03] A. HERTZMANN. « A Survey of Stroke-Based Rendering ». *IEEE Comp. Graphics and Applications*, vol. 23, pages 70–81, 2003. ISSN 0272-1716.
- [HESS06] C. HESS, P. MUKHERJEE, E. HAN, D. XU & D. VIGNERON. « Q-ball reconstruction of multimodal fiber orientations using the spherical harmonic basis. ». *Magn. Res. Med.*, 56(1), pages 104–117, 2006.
- [HIRSCH07] H. HIRSCHMÜLLER & D. SCHARSTEIN. « Evaluation of cost functions for stereo matching ». Dans *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8, 2007.
- [HSU94] S. C. HSU & I. H. H. LEE. « Drawing and animation using skeletal strokes ». Dans *SIGGRAPH '94 : Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 109–118, 1994. ISBN 0-89791-667-0.
- [HUZINA65] S. HUZINAGA. « Gaussian-Type Functions for Polyatomic Systems. I ». *Journal of Chemical Physics*, vol. 42, pages 1293–1302, 1965.
- [IIZUKA17] S. IIZUKA, E. SIMO-SERRA & H. ISHIKAWA. « Globally and locally consistent image completion ». *ACM Transactions on Graphics (TOG)*, 36(4), page 107, 2017.
- [JIA03] J. JIA & C.-K. TANG. « Image repairing : Robust image synthesis by adaptive nd tensor voting ». Dans *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 1, pages I–I, IEEE, 2003.
- [JIA04] J. JIA & C.-K. TANG. « Inference of segmented color and texture description by tensor voting ». *IEEE Trans. Pattern Anal.*, 26(6), pages 771–786, 2004.
- [JIAN07] B. JIAN, B. C. VEMURI, E. ÖZARSLAN, P. R. CARNEY & T. H. MARECI. « A novel tensor distribution model for the diffusion-weighted MR signal ». *NeuroImage*, vol. 37, pages 164–176, 2007.
- [JING17] Y. JING, Y. YANG, Z. FENG, J. YE & M. SONG. « Neural Style Transfer : A Review ». *CoRR*, vol. abs/1705.04058, 2017.

- [JOBARD97] B. JOBARD & W. LEFER. « Creating Evenly-Spaced Streamlines of Arbitrary Density ». Dans *Proc. 8th Eurographics Workshop on Visualization in Scientific Computing*, pages 45–55, 1997.
- [JOHNSO16] J. JOHNSON, A. ALAHI & L. FEI-FEI. « Perceptual losses for real-time style transfer and super-resolution ». Dans *European Conference on Computer Vision*, pages 694–711, Springer, 2016.
- [JONES04] D. JONES & P. BASSER. « Squashing peanuts and smashing pumpkins" : How noise distorts diffusion-weighted MR data ». *Magn. Res. Med.*, 52(5), pages 979–993, 2004.
- [JUNGEN81] M. JUNGEN. « A Gaussian basis for Rydberg orbitals ». *The Journal of Chemical Physics*, vol. 74, page 750, 1981.
- [KANATA93] K. KANATANI. « Geometric computation for machine vision ». Oxford University Press, Inc., 1993.
- [KANG07] H. KANG, S. LEE & C. K. CHUI. « Coherent line drawing ». Dans *NPAR '07 : Proceedings of the 5th international symposium on Non-photorealistic animation and rendering*, pages 43–50, 2007. ISBN 978-1-59593-624-0.
- [KANG09] H. KANG, S. LEE & C. K. CHUI. « Flow-Based Image Abstraction ». *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, pages 62–76, 2009. ISSN 1077-2626.
- [KANIZS79] G. KANIZSA & G. KANIZSA. « Organization in vision : Essays on Gestalt perception ». Praeger New York, 1979.
- [KAUFMA89] L. KAUFMAN & D. KRAMER. « Measuring Signal-to-Noise Ratios In MR Imaging ». *Radiology*, vol. 173, pages 265–267, 1989.
- [KAWAGI03] Y. KAWAGISHI, K. HATSUYAMA & K. KONDO. « Cartoon Blur : Non-Photorealistic Motion Blur ». *Computer Graphics International Conference*, vol. 0, page 276, 2003. ISSN 1530-1052.
- [KAWAI09] N. KAWAI, T. SATO & N. YOKOYA. « Image Inpainting Considering Brightness Change and Spatial Locality of Textures and Its Evaluation ». Dans *Proc. of the 3rd PSIVT*, pages 271–282, 2009. ISBN 978-3-540-92956-7.
- [KERVRA06] C. KERVRAAN & J. BOULANGER. « Optimal spatial adaptation for patch-based image denoising ». *IEEE Transactions on Image Processing*, 15(10), pages 2866–2878, 2006.
- [KHACHA07] M. KHACHATURIAN, J. WISCO & D. TUCH. « Boosting the sampling efficiency of q-Ball imaging using multiple wavevector fusion ». *Magn. Res. Med.*, vol. 57, pages 289–296, 2007.
- [KICHEN97] S. KICHENASSAMY. « The perona–malik paradox ». *SIAM Journal on Applied Mathematics*, 57(5), pages 1328–1342, 1997.
- [KIMMEL00] R. KIMMEL, R. MALLADI & N. SOCHEN. « Images as embedded maps and minimal surfaces : movies, color, texture, and volumetric medical images ». *International Journal of Computer Vision*, 39(2), pages 111–129, 2000.
- [KIMMEL02] R. KIMMEL & N. SOCHEN. « Orientation diffusion or how to comb a porcupine ». *Journal of Visual Communication and Image Representation*, 13(1-2), pages 238–248, 2002.
- [KINDER05] S. KINDERMANN, S. OSHER & P. W. JONES. « Deblurring and denoising of images by nonlocal functionals ». *Multiscale Modeling & Simulation*, 4(4), pages 1091–1115, 2005.
- [KING97] M. KING, J. HOUSEMAN, D. GADIAN & A. CONNELLY. « Localized q-space imaging of the mouse brain ». *Magn. Res. Med.*, 38(6), pages 930–937, 1997.
- [KING13] E. J. KING, G. KUTYNIOK & W.-Q. LIM. « Image inpainting : theoretical analysis and comparison of algorithms ». Dans *SPIE Optical Engineering+ Applications*, pages 885802–885802, International Society for Optics and Photonics, 2013.

- [KOENDE84] J. J. KOENDERINK. « The structure of images ». *Biological cybernetics*, 50(5), pages 363–370, 1984.
- [KOMODA06] N. KOMODAKIS. « Image completion using global optimization ». Dans *Computer Vision and Pattern Recognition*, vol. 1, pages 442–452, IEEE, 2006.
- [KOMODA07] N. KOMODAKIS & G. TZIRITAS. « Image completion using efficient belief propagation via priority scheduling and dynamic pruning ». *Image Processing, IEEE Transactions on*, 16(11), pages 2649–2661, 2007.
- [KORMAN11] S. KORMAN & S. AVIDAN. « Coherency sensitive hashing ». Dans *ICCV*, pages 1607–1614, IEEE, 2011.
- [KORNPR97A] P. KORNPORST, R. DERICHE & G. AUBERT. « Image restoration via PDE ». Dans *Enabling Technologies for Law Enforcement and Security*, pages 22–33, International Society for Optics and Photonics, 1997.
- [KORNPR97B] P. KORNPORST, R. DERICHE & G. AUBERT. « Nonlinear operators in image restoration ». Dans *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 325–330, IEEE, 1997.
- [KOSCHA95] A. KOSCHAN. « A comparative study on color edge detection ». Dans *Proceedings of the 2nd Asian Conference on Computer Vision*, vol. 3, pages 574–578, 1995.
- [KUCHEL97] P. KUCHEL, A. COY & P. STILBS. « NMR diffusion-diffraction of water revealing alignment of erythrocytes in a magnetic field and their dimensions and membrane transport characteristics ». *Magn. Res. Med.*, 37(5), pages 637–643, 1997.
- [KUCHEL99] P. W. KUCHEL & C. J. DURRANT. « Permeability Coefficients from NMR q-Space Data : Models with Unevenly Spaced Semi-permeable Parallel Membranes ». *Journal of Magnetic Resonance*, vol. 139, pages 258–272, 1999.
- [KUYPER06] T. KUYPER. « Luminosity masks ». <http://goodlight.us/writing/luminositymasks/luminositymasks-1.html>, 2006.
- [LAIDLA09] D. H. LAIDLAW & J. WEICKERT. « Visualization and Processing of Tensor Fields : Advances and Perspectives ». Springer Science & Business Media, 2009.
- [LAURIE49] A. P. LAURIE. « The technique of the great painters ». Carrol & Nicholson, 1949.
- [LE BIH88] D. LE BIHAN, E. BRETON & D. LALLEMAND. « Perfusion in Intravoxel Incoherent Motion MR Imaging ». *Radiology*, vol. 168, pages 497–505, 1988.
- [LE MEU11] O. LE MEUR, J. GAUTIER & C. GUILLEMOT. « Exemplar-based inpainting based on local geometry ». Dans *ICIP*, pages 3401–3404, Brussel, Belgium, 2011.
- [LE MEU13A] O. LE MEUR, M. EBDELLI & C. GUILLEMOT. « Hierarchical Super-Resolution-Based Inpainting ». *IEEE T. Image Process.*, 22(10), pages 3779–3790, 2013. ISSN 1057-7149.
- [LE MEU13B] O. LE MEUR, M. EBDELLI & C. GUILLEMOT. « Hierarchical super-resolution-based inpainting ». *Image Processing, IEEE Transactions on*, 22(10), pages 3779–3790, 2013.
- [LEBIHA86] D. LEBIHAN, E. BRETON, D. LALLEMAND, P. GRENIER, E. CABANIS & M. LAVAL-JEANTET. « MR imaging of intravoxel incoherent motions : Application to diffusion and perfusion in neurologic disorders ». *Radiology*, pages 401–407, 1986.
- [LECUN95] Y. LECUN, Y. BENGIO ET AL. « Convolutional networks for images, speech, and time series ». *The handbook of brain theory and neural networks*, 3361(10), page 1995, 1995.
- [LECUN98] Y. LECUN, L. BOTTOU, Y. BENGIO & P. HAFFNER. « Gradient-based learning applied to document recognition ». *Proceedings of the IEEE*, 86(11), pages 2278–2324, 1998.
- [LEEDAN00] Y. LEEDAN & P. MEER. « Heteroscedastic regression in computer vision : Problems with bilinear constraint ». *International Journal of Computer Vision*, 37(2), pages 127–150, 2000.

- [LEVIN03] A. LEVIN, A. ZOMET & Y. WEISS. « Learning how to inpaint from global image statistics ». Dans *ICCV*, pages 305–312, IEEE, 2003.
- [LIBOVE80] J. LIBOVE & J. SINGER. « Resolution and signal-to-noise relationships in NMR imaging in the human body ». *Journal of Physics E : Scientific Instruments*, vol. 13, pages 38–44, 1980.
- [LINDEB13] T. LINDBERG. « Scale-space theory in computer vision », vol. 256. Springer Science & Business Media, 2013.
- [LITWIN97] P. LITWINOWICZ. « Processing images and video for an impressionist effect ». Dans *SIGGRAPH '97 : Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 407–414, 1997. ISBN 0-89791-896-7.
- [LIU04] C. LIU, R. BAMMER, B. ACAR & M. MOSELEY. « Characterizing non-Gaussian diffusion by using generalized diffusion tensors ». *Magn. Res. Med.*, vol. 51, pages 924–937, 2004.
- [LUCAS81] B. D. LUCAS, T. KANADE ET AL. « An iterative image registration technique with an application to stereo vision ». Dans *Imaging Understanding Workshop*, pages 121–130, Vancouver, BC, Canada, 1981.
- [LUSTIG07] M. LUSTIG, D. DONOHO & J. M. PAULY. « Sparse MRI : The application of compressed sensing for rapid MR imaging ». *Magn. Res. Med.*, 58(6), pages 1182–1195, 2007.
- [MA12] L. MA, L. DO & P. DE WITH. « Depth-guided inpainting algorithm for Free-Viewpoint Video ». Dans *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 1721–1724, 2012.
- [MAIRAL08] J. MAIRAL, M. ELAD & G. SAPIRO. « Sparse representation for color image restoration ». *IEEE T. Image Process.*, 17(1), pages 53–69, 2008.
- [MANSFI11] A. MANSFIELD, M. PRASAD, C. ROTHER, T. SHARP, P. KOHLI & L. J. VAN GOOL. « Transforming Image Completion. ». Dans *BMVC*, pages 1–11, 2011.
- [MAO13] Y. MAO, G. CHEUNG, A. ORTEGA & Y. JI. « Expansion hole filling in depth-image-based rendering using graph-based interpolation ». Dans *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 1859–1863, IEEE, 2013.
- [MAO14] Y. MAO, G. CHEUNG & Y. JI. « Image interpolation for DIBR view synthesis using graph fourier transform ». Dans *3DTV-Conference : The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON), 2014*, pages 1–4, IEEE, 2014.
- [MARKOS97] L. MARKOSIAN, M. A. KOWALSKI, D. GOLDSTEIN, S. J. TRYCHIN, J. F. HUGHES & L. D. BOURDEV. « Real-time nonphotorealistic rendering ». Dans *SIGGRAPH '97 : Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 415–420, 1997. ISBN 0-89791-896-7.
- [MARKOS00] L. MARKOSIAN, B. J. MEIER, M. A. KOWALSKI, L. S. HOLDEN, J. D. NORTHRUP & J. F. HUGHES. « Art-based rendering with continuous levels of detail ». Dans *NPAR '00 : Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, pages 59–66, 2000. ISBN 1-58113-277-8.
- [MARTIN12] R. MARTINEZ-NORIEGA, A. ROUMY & G. BLANCHARD. « Exemplar-based image inpainting : Fast priority and coherent nearest neighbor search ». Dans *MLSP*, pages 1–6, IEEE, 2012.
- [MASNOU98] S. MASNOU & J.-M. MOREL. « Level lines based disocclusion ». Dans *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, pages 259–263, IEEE, 1998.

- [MASNOU02] S. MASNOU. « Disocclusion : a variational approach using level lines ». *IEEE Trans. Image Process.*, 11(2), pages 68–76, 2002.
- [MEDION00] G. MEDIONI, M.-S. LEE & C.-K. TANG. « A computational framework for segmentation and grouping ». Elsevier, 2000.
- [MEIER96] B. J. MEIER. « Painterly rendering for animation ». Dans *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 477–484, 1996. ISBN 0-89791-746-4.
- [MEYER01] Y. MEYER. « Oscillating patterns in image processing and nonlinear evolution equations : the fifteenth Dean Jacqueline B. Lewis memorial lectures », vol. 22. American Mathematical Soc., 2001.
- [MORRIS86] P. MORRIS. « Nuclear Magnetic Resonance Imaging in Medicine and Biology ». Oxford University Press, USA, 1986.
- [MOSELE90] M. MOSELEY, Y. COHEN, J. MINTOROVITCH, L. CHILEUITT, H. SHIMIZU, J. KUCHARCZYK, M. WENDLAND & P. WEINSTEIN. « Early detection of regional cerebral ischemia in cats : comparison of diffusion and T 2-weighted MRI and spectroscopy ». *Magn. Res. Med.*, 14(2), pages 330–346, 1990.
- [MUDDAL14] S. M. MUDDALA, M. SJOSTROM & R. OLSSON. « Depth-based inpainting for disocclusion filling ». Dans *3DTV-Conference : The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4, 2014.
- [MULKER00] R. MULKERN, H. ZENGINONUL, R. ROBERTSON, P. BOGNER, K. ZOU, H. GUDBJARTSSON, C. GUTTMANN, D. HOLTZMAN, W. KYRIAKOS, F. JOLESZ ET AL. « Multi-component apparent diffusion coefficients in human brain : Relationship to spin-lattice relaxation ». *Magn. Res. Med.*, 44(2), pages 292–300, 2000.
- [MURPHY93] B. MURPHY, P. CARSON, J. ELLIS, Y. ZHANG, R. HYDE & T. CHENEVERT. « Signal-to-noise measures for magnetic resonance imagers ». *Magnetic Resonance Imaging*, 11(3), pages 425–428, 1993.
- [NDJIKI11] P. NDJIKI-NYA, M. KOPPEL, D. DOSHKOV, H. LAKSHMAN, P. MERKLE, K. MULLER & T. WIEGAND. « Depth image-based rendering with advanced texture synthesis for 3-D video ». *Multimedia, IEEE Transactions on*, 13(3), pages 453–465, 2011.
- [NEWSON13] A. NEWSON, M. FRADET, P. PÉREZ, A. ALMANSA & Y. GOUSSEAU. « Towards fast, generic video inpainting ». Dans *European Conference on Visual Media Production, 2013. CVMP 2013.*, pages 1–8, 2013.
- [NIELSE97] M. NIELSEN, L. FLORACK & R. DERICHE. « Regularization, scale-space, and edge detection filters ». *Journal of Mathematical Imaging and Vision*, 7(4), pages 291–307, 1997.
- [NIENDO96] T. NIENDORF, R. DIJKHUIZEN, N. D.G., C. M. VAN LOOKEREN & K. NICOLAY. « Biexponential diffusion attenuation in various states of brain tissue : implications for diffusion-weighted imaging ». *Magn. Res. Med.*, vol. 36, pages 847–857, 1996.
- [NIKOLO01] M. NIKOLOVA. « Image restoration by minimizing objective functions with nonsmooth data-fidelity terms ». Dans *Variational and Level Set Methods in Computer Vision, 2001. Proceedings. IEEE Workshop on*, pages 11–18, IEEE, 2001.
- [NORTH00] J. D. NORTHRUP & L. MARKOSIAN. « Artistic silhouettes : a hybrid approach ». Dans *NPAR '00 : Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, pages 31–37, 2000. ISBN 1-58113-277-8.
- [ONCU12] A. I. ONCU, F. DEGER & J. Y. HARDEBERG. « Evaluation of digital inpainting quality in the context of artwork restoration ». Dans *European Conference on Computer Vision*, pages 561–570, Springer, 2012.

- [OSHER90] S. OSHER & L. I. RUDIN. « Feature-oriented image enhancement using shock filters ». *SIAM Journal on Numerical Analysis*, 27(4), pages 919–940, 1990.
- [OSTROM99] V. OSTROMOUKHOV. « Digital facial engraving ». Dans *SIGGRAPH '99 : Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 417–424, 1999. ISBN 0-201-48560-5.
- [OZARSL03] E. OZARSLAN & T. MARECI. « Generalized diffusion tensor imaging and analytical relationships between diffusion tensor imaging and high angular resolution diffusion imaging ». *Magn. Res. Med.*, 50(5), pages 955–965, 2003.
- [ÖZAR06] E. ÖZARSLAN, T. M. SHERPERD, B. C. VEMURI, S. J. BLACKBAND & T. H. MARECI. « Resolution of complex tissue microarchitecture using the diffusion orientation transform (DOT) ». *NeuroImage*, vol. 31, pages 1086–1103, 2006.
- [ÖZAR07] E. ÖZARSLAN & P. J. BASSER. « MR diffusion-‘diffraction’ phenomenon in multi-pulse-field-gradient experiments ». *Journal of Magnetic Resonance*, 188(2), pages 285–294, 2007.
- [ÖZAR08] E. ÖZARSLAN, C. KOAY & P. BASSER. « Simple harmonic oscillator based estimation and reconstruction for one-dimensional q-space MR ». Dans *Proc. Intl. Soc. Mag. Reson. Med*, vol. 16, page 35, 2008.
- [PAPYAN16] V. PAPYAN, Y. ROMANO & M. ELAD. « Convolutional neural networks analyzed via convolutional sparse coding ». *stat*, vol. 1050, page 27, 2016.
- [PARIS06] S. PARIS & F. DURAND. « A fast approximation of the bilateral filter using a signal processing approach ». Dans *European conference on computer vision*, pages 568–580, Springer, 2006.
- [PENNEC06] X. PENNEC, P. FILLARD & N. AYACHE. « A Riemannian framework for tensor computing ». *International Journal of Computer Vision*, 66(1), pages 41–66, 2006.
- [PÉRE03] P. PÉREZ, M. GANGNET & A. BLAKE. « Poisson image editing ». *ACM T. Graphic.*, 22(3), pages 313–318, juillet 2003. ISSN 0730-0301.
- [PÉRE04] P. PÉREZ, M. GANGNET & A. BLAKE. « PatchWorks : Example-Based Region Tiling for Image Editing ». Rapport technique, Microsoft Research, MSR-TR-2004-04, 2004.
- [PERONA90] P. PERONA & J. MALIK. « Scale-space and edge detection using anisotropic diffusion ». *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 12(7), pages 629–639, 1990.
- [PERONA98] P. PERONA. « Orientation diffusion ». *IEEE Transactions on Image Processing*, 7(3), pages 457–467, 1998.
- [PEYR09] G. PEYRÉ. « Manifold models for signals and images ». *Computer Vision and Image Understanding*, 113(2), pages 249–260, 2009.
- [PEYR10] G. PEYRÉ. « Texture Synthesis with Grouplets ». *IEEE Trans. Pattern Anal.*, 4(32), pages 733–746, 2010.
- [PICKAL06] V. PICKALOV & P. J. BASSER. « 3-D tomographic reconstruction of the average propagator from MRI data ». Dans *Biomedical Imaging : Nano to Macro, 2006. 3rd IEEE International Symposium on*, pages 710–713, IEEE, 2006.
- [POYNT095] C. POYNTON. « Poynton’s colour FAQ ». www.inforamp.net/poynton, 1995.
- [PRAUN01] E. PRAUN, H. HOPPE, M. WEBB & A. FINKELSTEIN. « Real-time hatching ». Dans *SIGGRAPH '01 : Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, page 581, 2001. ISBN 1-58113-374-X.
- [PRESS92] W. PRESS, B. FLANNERY, S. TEUKOLSKY & W. VETTERLING. « Numerical Recipes in FORTRAN : The Art of Scientific Computing », chapitre Runge-Kutta Method, pages 704–716. Cambridge University Press, 1992.

- [PRITCH09] Y. PRITCH, E. KAV-VENAKI & S. PELEG. « Shift-map image editing ». Dans *International Conference on Computer Vision*, pages 151–158, IEEE, 2009.
- [QI13] F. QI, J. HAN, P. WANG, G. SHI & F. LI. « Structure guided fusion for depth map inpainting ». *Pattern Recognition Letters*, 34(1), pages 70–76, 2013.
- [RABIN14] J. RABIN, S. FERRADANS & N. PAPADAKIS. « Adaptive Color Transfer With Relaxed Optimal Transport ». Dans *IEEE international Conference on Image Processing*, page 5, Paris, France, octobre 2014. 5 pages.
- [RAMIRE03] A. RAMIREZ-MANZANARES, M. RIVERA ET AL. « Brain nerve bundles estimation by restoring and filtering intra-voxel information in diffusion tensor MRI ». *Procc*, vol. 2, pages 73–80, 2003.
- [REEL13] S. REEL, G. CHEUNG, P. WONG & L. S. DOOLEY. « Joint texture-depth pixel inpainting of disocclusion holes in virtual view synthesis ». Dans *Signal and Information Processing Association Annual Summit and Conference*, pages 1–7, 2013.
- [REINHA01] E. REINHARD, M. ADHIKMIN, B. GOOCH & P. SHIRLEY. « Color transfer between images ». *IEEE Computer graphics and applications*, 21(5), pages 34–41, 2001.
- [RITCHI98] D. RITCHIE. « Parametric Protein Shape Recognition ». Thèse de doctorat, University of Aberdeen, U.K., 1998.
- [RITCHI05] D. W. RITCHIE. « High-order analytic translation matrix elements for real-space six-dimensional polar Fourier correlations ». *J. Appl. Cryst.*, vol. 38, pages 808–818, 2005.
- [ROTH05] S. ROTH & M. J. BLACK. « Fields of experts : A framework for learning image priors ». Dans *CVPR*, vol. 2, pages 860–867, IEEE, 2005.
- [RUDIN92] L. I. RUDIN, S. OSHER & E. FATEMI. « Nonlinear total variation based noise removal algorithms ». *Physica D : Nonlinear Phenomena*, 60(1-4), pages 259–268, 1992.
- [SALISB96] M. SALISBURY, C. ANDERSON, D. LISCHINSKI & D. SALESIN. « Scale-dependent reproduction of pen-and-ink illustrations ». Dans *SIGGRAPH*, pages 461–468, 1996.
- [SALISB97] M. P. SALISBURY, M. T. WONG, J. F. HUGHES & D. H. SALESIN. « Orientable textures for image-based pen-and-ink illustration ». Dans *SIGGRAPH '97 : Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 401–406, 1997. ISBN 0-89791-896-7.
- [SAPIRO96A] G. SAPIRO. « Vector-Valued Active Contours ». Dans *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 680–685, 1996.
- [SAPIRO96B] G. SAPIRO & D. RINGACH. « Anisotropic diffusion of multivalued images with applications to color filtering ». *IEEE Transactions on Image Processing*, 5(11), pages 1582–1585, 1996.
- [SAPIRO97] G. SAPIRO. « Color Snakes ». *Computer Vision and Image Understanding*, 68(2), 1997.
- [SAPIRO01] G. SAPIRO. « Geometric partial differential equations and image analysis ». Cambridge university press, 2001.
- [SAUVAG08] C. SAUVAGET & V. BOYER. « Comics Stylization from Photographs ». Dans *ISVC '08 : Proceedings of the 4th International Symposium on Advances in Visual Computing*, pages 1125–1134, 2008. ISBN 978-3-540-89638-8.
- [SCHARS03] D. SCHARSTEIN & R. SZELISKI. « High-accuracy stereo depth maps using structured light ». Dans *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 1, pages I–195, 2003.
- [SCHARS07] D. SCHARSTEIN & C. PAL. « Learning conditional random fields for stereo ». Dans *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8, 2007.

- [SCHARS14] D. SCHARSTEIN, H. HIRSCHMÜLLER, Y. KITAJIMA, G. KRATHWOHL, N. NEŠIĆ, X. WANG & P. WESTLING. « High-resolution stereo datasets with subpixel-accurate ground truth ». Dans *Pattern Recognition*, pages 31–42. Springer, 2014.
- [SCHMEI15] M. SCHMEING & J. XIAOYI. « Faithful Disocclusion Filling in Depth Image Based Rendering Using Superpixel-Based Inpainting ». *Multimedia, IEEE transactions on*, 17(12), pages 2160–2173, 2015.
- [SECORD00] A. SECORD. « Random Marks on Paper : Non-Photorealistic Rendering with Small Primitives », 2000.
- [SHAH96] J. SHAH. « Curve evolution and segmentation functionals : Applications to color images ». Dans *Proceedings of the International Conference on Image Processing (ICIP)*, pages 461–464, 1996.
- [SHEN01] J. SHEN & T. CHAN. « Variational restoration of nonflat image features : Models and algorithms ». *SIAM Journal on Applied Mathematics*, 61(4), pages 1338–1361, 2001.
- [SHEN02] J. SHEN & T. F. CHAN. « Mathematical models for local nontexture inpaintings ». *SIAM J. Appl. Math.*, 62(3), pages 1019–1043, 2002.
- [SHEN03] J. SHEN, S. H. KANG & T. F. CHAN. « Euler’s elastica and curvature-based inpainting ». *SIAM Journal on Applied Mathematics*, 63(2), pages 564–592, 2003.
- [SIJBER98] J. SIJBERS, A. DEN DEKKER, J. VAN AUDEKERKE, M. VERHOYE & D. VAN DYCK. « Estimation of the noise in magnitude mr images ». *Magnetic Resonance Imaging*, vol. 16, pages 87–90, 1998.
- [SIJBER07] J. SIJBERS, D. POOT, A. DEKKER & W. PINTJENS. « Automatic estimation of the noise variance from the histogram of a magnetic resonance image ». *Physics in Medecine and Biology*, 52(5), page 1335, 2007.
- [SIMAKO08] D. SIMAKOV, Y. CASPI, E. SHECHTMAN & M. IRANI. « Summarizing visual data using bidirectional similarity ». Dans *CVPR*, pages 1–8, IEEE, 2008.
- [SIMONY14] K. SIMONYAN & A. ZISSERMAN. « Very Deep Convolutional Networks for Large-Scale Image Recognition ». *CoRR*, vol. abs/1409.1556, 2014.
- [SOCHEN98A] N. SOCHEN, R. KIMMEL & R. MALLADI. « A geometrical framework for low level vision ». *IEEE Transaction on Image Processing, Special Issue on PDE based Image Processing*, 7(3), pages 310–318, 1998.
- [SOCHEN98B] N. SOCHEN & Y. Y. ZEEVI. « Representation of colored images by manifolds embedded in higher dimensional non-Euclidean space ». Dans *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, vol. 1, pages 166–170, IEEE, 1998.
- [SOCHEN01A] N. SOCHEN. « On affine invariance in the Beltrami framework for vision ». Dans *IEEE Workshop on Variational and Level Set Methods (VLSM)*, pages 51–56, 2001.
- [SOCHEN01B] N. SOCHEN, R. KIMMEL & A. BRUCKSTEIN. « Diffusions and confusions in signal and image processing ». *Journal of Mathematical Imaging and Vision (JMIV)*, 14(3), pages 195–209, 2001.
- [SOILLE13] P. SOILLE. « Morphological image analysis : principles and applications ». Springer Science & Business Media, 2013.
- [SOKOLN51] I. S. SOKOLNIKOFF. « Tensor analysis : Theory and applications ». Wiley, 1951.
- [STALLI95] D. STALLING & H. HEGE. « Fast and Resolution Independent Line Integral Convolution ». Dans *ACM SIGGRAPH, 22nd Annual Conference on Computer Graphics and Interactive Technique*, pages 249–256, 1995.
- [STARCK05] J.-L. STARCK, M. ELAD & D. L. DONOHO. « Image decomposition via the combination of sparse representations and a variational approach ». *IEEE T. Image Process.*, 14(10), pages 1570–1582, 2005.

- [STEJSK65] E. STEJSKAL & J. TANNER. « Spin diffusion measurements : spin echoes in the presence of a time-dependent field gradient ». *Journal of Chemical Physics*, vol. 42, pages 288–292, 1965.
- [SU94] Z. SU & P. COPPENS. « Rotation of real spherical harmonics ». *Foundations of Crystallography*, 50(5), page 7673, 1994.
- [SULAM17] J. SULAM, V. POPYAN, Y. ROMANO & M. ELAD. « Multi-Layer Convolutional Sparse Modeling : Pursuit and Dictionary Learning ». *arXiv preprint arXiv :1708.08705*, 2017.
- [SUN05] J. SUN, L. YUAN, J. JIA & H.-Y. SHUM. « Image completion with structure propagation ». *ACM T. Graphic.*, 24(3), pages 861–868, juillet 2005. ISSN 0730-0301.
- [SZAFA95] A. SZAFAER, J. ZHONG & J. C. GORE. « Theoretical model for water diffusion in tissues ». *Magn. Res. Med.*, vol. 33, pages 697–712, 1995.
- [TANG98] B. TANG, G. SAPIRO & V. CASELLES. « Direction diffusion ». Dans *International Conference on Computer Vision (ICCV)*, 1998.
- [TANG00] B. TANG, G. SAPIRO & V. CASELLES. « Diffusion of general data on non-flat manifolds via harmonic maps theory : The direction diffusion case ». *International Journal of Computer Vision (IJCV)*, 36(2), pages 149–161, 2000.
- [TANNER68] J. TANNER & E. STEJSKAL. « Restricted Self-Diffusion of Protons in Colloidal Systems by the Pulsed-Gradient, Spin-Echo Method ». *Journal of Chemical Physics*, vol. 49, pages 1768–1777, 1968.
- [TANNER78] J. TANNER. « Transient diffusion in a system partitioned by permeable barriers. Application to NMR measurements with a pulsed field gradient ». *Journal of Chemical Physics*, vol. 69, page 1748, 1978.
- [TEBOUL98] S. TEBOUL, L. BLANC-FÉRAUD, G. AUBERT & M. BARLAUD. « Variational approach for edge-preserving regularization using coupled PDE's ». *IEEE Transaction on Image Processing, Special Issue on PDE based Image Processing*, 7(3), pages 387–397, 1998.
- [THOMPS56] D. V. THOMPSON. « The materials and techniques of medieval painting », vol. 327. Courier Corporation, 1956.
- [TIAN09] D. TIAN, P.-L. LAI, P. LOPEZ & C. GOMILA. « View synthesis techniques for 3D video ». Dans *SPIE Optical Engineering+ Applications*, pages 74430T–74430T, International Society for Optics and Photonics, 2009.
- [TIKHON63] A. N. TIKHONOV. « Regularization of incorrectly posed problems ». *SOVIET MATHEMATICS DOKLADY*, 1963.
- [TOLBA99] O. TOLBA, J. DORSEY & L. McMILLAN. « Sketching with projective 2D strokes ». Dans *UIST '99 : Proceedings of the 12th annual ACM symposium on User interface software and technology*, pages 149–157, 1999. ISBN 1-58113-075-9.
- [TOMASI98] C. TOMASI & R. MANDUCHI. « Bilateral filtering for gray and color images ». Dans *Computer Vision, 1998. Sixth International Conference on*, pages 839–846, IEEE, 1998.
- [TOURNI04] J. TOURNIER, F. CALAMANTE, D. GADIAN & A. CONNELLY. « Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution ». *NeuroImage*, vol. 23, pages 1179–1185, 2004.
- [TSCHUM00] D. TSCHUMPERLÉ. « CImg : The C++ Template Image Processing Library ». <http://cimg.eu>, 2000.
- [TSCHUM01] D. TSCHUMPERLÉ & R. DERICHE. « Diffusion tensor regularization with constraints preservation ». Dans *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, pages I–I, IEEE, 2001.

- [TSCHUM02] D. TSCHUMPERLÉ & R. DERICHE. « Diffusion PDE's on vector-valued images : local approach and geometric viewpoint ». *IEEE Signal Processing Magazine*, 19(5), pages 16–25, 2002.
- [TSCHUM03] D. TSCHUMPERLÉ & R. DERICHE. « Tensor Field Visualization with PDE's and Application to DT-MRI Fiber Visualization ». Dans *IEEE Workshop on Variational and Level Set Methods*, 2003.
- [TSCHUM05] D. TSCHUMPERLE & R. DERICHE. « Vector-valued image regularization with PDEs : A common framework for different applications ». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4), pages 506–517, 2005.
- [TSCHUM06] D. TSCHUMPERLÉ. « Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's ». *International Journal of Computer Vision*, 68(1), pages 65–82, 2006.
- [TSCHUM07] D. TSCHUMPERLE & R. DERICHE. « Anisotropic diffusion partial differential equations for multichannel image regularization : Framework and applications ». *Advances in Imaging and Electron Physics*, vol. 145, pages 149–209, 2007.
- [TSCHUM09] D. TSCHUMPERLÉ & L. BRUN. « Non-local image smoothing by applying anisotropic diffusion PDE's in the space of patches ». Dans *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 2957–2960, IEEE, 2009.
- [TSCHUM11] D. TSCHUMPERLÉ. « Tensor-directed simulation of strokes for image stylization with hatching and contours ». Dans *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 793–796, IEEE, 2011.
- [TUCH99] D. S. TUCH, R. WEISSKOFF, J. BELLIVEAU & V. WEDEEN. « High angular resolution diffusion imaging of the human brain ». Dans *Proceedings of the 7th Annual Meeting of ISMRM, Philadelphia*, vol. 321, 1999.
- [TUCH04] D. TUCH. « Q-ball imaging ». *Magn. Res. Med.*, vol. 52, pages 1358–1372, 2004.
- [TURK96] G. TURK & D. BANKS. « Image-guided streamline placement ». Dans *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 453–460, 1996. ISBN 0-89791-746-4.
- [ULYANO16] D. ULYANOV, V. LEBEDEV, A. VEDALDI & V. S. LEMPITSKY. « Texture Networks : Feed-forward Synthesis of Textures and Stylized Images. ». Dans *ICML*, pages 1349–1357, 2016.
- [WANG04] Z. WANG, B. VEMURI, Y. CHEN & T. MARECI. « A constrained variational principle for direct estimation and smoothing of the diffusion tensor field from complex DWI ». *Transactions on Medical Imaging*, pages 930–939, 2004.
- [WEDEEN00] V. WEDEEN, T. REESE, D. TUCH, M. WEIGEL, J. DOU, R. WEISKOFF & D. CHESSLER. « Mapping fiber orientation spectra in cerebral white matter with Fourier-transform diffusion MRI ». Dans *Proceedings of the 8th Annual Meeting of ISMRM, Denver*, page 82, 2000.
- [WEI00] L.-Y. WEI & M. LEVOY. « Fast texture synthesis using tree-structured vector quantization ». Dans *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 479–488, ACM Press/Addison-Wesley Publishing Co., 2000.
- [WEICKE97A] J. WEICKERT. « A review of nonlinear diffusion filtering ». Dans B. SPRINGER, rédacteur, *Scale-Space Theory in Computer Vision, Lecture Notes in Computer Science*, vol. 1252, pages 3–28, 1997.
- [WEICKE97B] J. WEICKERT & B. BENHAMOUDA. « Why the Perona-Malik filter works ». Rapport technique, Technical Report 97/22, Department of Computer Science, University of Copenhagen, 1997.

- [WEICKE98] J. WEICKERT. « Anisotropic diffusion in image processing », vol. 1. Teubner Stuttgart, 1998.
- [WEICKE99A] J. WEICKERT. « Coherence-Enhancing Diffusion Filtering ». *Int. Journal. Comput. Vision*, vol. 31, pages 111–127, April 1999.
- [WEICKE99B] J. WEICKERT. « Coherence-Enhancing Diffusion of Colour Images ». *Image and Vision Computing*, vol. 17, pages 199–210, 1999.
- [WEICKE02] J. WEICKERT & T. BROX. « Diffusion and Regularization of Vector and Matrix-valued Images ». *Inverse Problems, Image Analysis, and Medical Imaging*, vol. 313, pages 251–268, 2002.
- [WEICKE03] J. WEICKERT. « Coherence-enhancing shock filters ». Dans *Joint Pattern Recognition Symposium*, pages 1–8, Springer, 2003.
- [WELSH02] T. WELSH, M. ASHIKHMIN & K. MUELLER. « Transferring color to greyscale images ». Dans *ACM Transactions on Graphics (TOG)*, 21(3), pages 277–280, ACM, 2002.
- [WEXLER04] Y. WEXLER, E. SHECHTMAN & M. IRANI. « Space-time video completion ». Dans *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1, pages I–120, IEEE, 2004.
- [WEXLER07] Y. WEXLER, E. SHECHTMAN & M. IRANI. « Space-Time Completion of Video ». *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3), pages 463–476, mars 2007. ISSN 0162-8828.
- [WIEST-08] N. WIEST-DAESSLE, S. PRIMA, P. COUPE, S. MORRISSEY & C. BARILLOT. « Rician noise removal by non-local means filtering for low signal-to-noise ratio MRI : Application to DT-MRI ». Dans *MICCAI*, pages 104–117, New York/USA, September 2008.
- [WINKEN94] G. WINKENBACH & D. SALESIN. « Computer-generated pen-and-ink illustration ». Dans *SIGGRAPH*, pages 91–100, 1994.
- [WINKEN96] G. WINKENBACH & D. H. SALESIN. « Rendering parametric surfaces in pen and ink ». Dans *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 469–476, 1996. ISBN 0-89791-746-4.
- [WITKIN83] A. WITKIN. « Scale-space filtering ». Dans *International Joint Conference on Artificial Intelligence*, pages 1019–1021, 1983.
- [WONG08] A. WONG & J. ORCHARD. « A nonlocal-means approach to exemplar-based inpainting ». Dans *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 2600–2603, IEEE, 2008.
- [WU07] Y.-C. WU & A. L. ALEXANDER. « Hybrid diffusion imaging ». *NeuroImage*, 36(3), pages 617 – 629, 2007.
- [XU10] Z. XU & J. SUN. « Image inpainting by patch propagation using patch sparsity ». *IEEE T. Image Process.*, 19(5), pages 1153–1165, 2010.
- [XU13] X. XU, L.-M. PO, C.-H. CHEUNG, L. FENG, K.-H. NG & K.-W. CHEUNG. « Depth-aided exemplar-based hole filling for DIBR view synthesis ». Dans *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, pages 2840–2843, 2013.
- [YABLON03] D. A. YABLONSKIY, G. L. BRETTHORST & J. J. ACKERMAN. « Statistical model for diffusion attenuated MR Signal ». *Magn. Res. Med.*, vol. 50, pages 664–669, 2003.
- [YEZZI98] A. YEZZI. « Modified curvature motion for image smoothing and enhancement ». *IEEE Transactions on Image Processing*, 7(3), pages 345–352, 1998.
- [YOON14] S. S. YOON, H. SOHN, Y. J. JUNG & Y. M. RO. « Inter-view consistent hole filling in view extrapolation for multi-view image generation ». Dans *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 2883–2887, IEEE, 2014.

- [ZHANG16] R. ZHANG, P. ISOLA & A. A. EFROS. « Colorful image colorization ». Dans *European Conference on Computer Vision*, pages 649–666, Springer, 2016.
- [ZHANG17] R. ZHANG, J.-Y. ZHU, P. ISOLA, X. GENG, A. S. LIN, T. YU & A. A. EFROS. « Real-time user-guided image colorization with learned deep priors ». *arXiv preprint arXiv :1705.02999*, 2017.
- [ZIVKOV68] T. ZIVKOVIC & Z. MAKSIC. « Explicit Formulas for Molecular Integrals over Hermite-Gaussian Functions ». *Journal of Chemical Physics*, vol. 49, pages 3083–3087, 1968.

Curriculum Vitae

David Tschumperlé

Né le 28/07/1976 (41 ans), Nationalité Française.

Pacé, deux enfants.

Web : <https://tschumperle.users.greyc.fr/>

E-Mail : David.Tschumperle@ensicaen.fr

Tél : 02.31.45.29.24



Diplômes - Qualifications

2004	<p>Qualifié pour les fonctions de Maître de Conférences en sections CNU 26 (Mathématiques Appliquées), 27 (Informatique) et 61 (Traitement du Signal).</p>
2002	<p>Thèse de doctorat de l'Université de Nice-Sophia Antipolis, mention Très Honorable.</p> <ul style="list-style-type: none"> <input type="checkbox"/> <i>Spécialité Automatique, Traitement du Signal et de l'Image.</i> <input type="checkbox"/> <i>Récompensée en 2003 par le 2ème prix de thèse de la Télécom Valley (région PACA).</i> <input type="checkbox"/> <i>Sélectionnée en 2003 comme l'une des 13 thèses européennes participant au prix Cor Baayen Award organisé par l'ERCIM.</i>
1999	<p>Diplôme d'Ingénieur de l'Ecole Supérieure en Sciences Informatiques (ESSI, devenue PolyTech'Nice-Sophia), Sophia Antipolis/France.</p> <p>DEA (Master Recherche) de l'Université de Nice, spécialité "Algorithmique, Robotique, Automatique, Vision, Image et Signal". Mention Bien.</p>

Parcours scientifique

A partir du 1er Octobre 2017	<p>Responsable de l'équipe IMAGE du laboratoire GREYC de Caen (UMR 6072), équipe composée de 5 PR, 11 MCF, 2 CR et 14 doctorants.</p>
Depuis le 1er Octobre 2004	<p>Chargé de recherche CNRS en section 07 (1ère classe, en 2009), affecté au laboratoire GREYC - Equipe IMAGE - CNRS UMR 6072 / ENSICAEN / Université de Caen.</p> <p><input type="checkbox"/> <i>Thème de recherche actuel : Méthodes variationnelles et non-locales pour le traitement d'images multi-dimensionnelles.</i></p>
Mars-Juillet 2004	<p>Ingénieur expert contractuel dans le cadre du projet européen PRES-TOSPACE, Université de La Rochelle, Equipe L31.</p> <p><input type="checkbox"/> <i>Désentrelacement de séquences vidéos par analyse de mouvement.</i></p>
Janvier - Décembre 2003	<p>Post-doctorat dans le projet ODYSSEE de l'INRIA Sophia-Antipolis (Institut National de Recherche en Informatique et Automatique).</p> <p><input type="checkbox"/> <i>Estimation, régularisation et visualisation d'images IRM de tenseurs de diffusion.</i></p> <p><input type="checkbox"/> <i>Collaboration avec le laboratoire IRMf de l'hôpital de la Timone (Marseille).</i></p>
Octobre 1999 - Décembre 2002	<p>Doctorat effectué dans le projet ROBOTVIS de l'INRIA Sophia-Antipolis. Directeur de thèse : Rachid Deriche.</p> <p><input type="checkbox"/> <i>Titre de la thèse : PDE-Based Regularization of Multivalued Images and Applications.</i></p>
Avril - Septembre 1999	<p>Stage de DEA effectué dans le projet ROBOTVIS de l'INRIA Sophia-Antipolis.</p> <p><input type="checkbox"/> <i>Sujet du stage : Diffusion anisotrope et images vectorielles.</i></p>

Informations complémentaires

Recherche académique	<p>Auteur de publications scientifiques en langues anglaise et française (<i>plus de 70</i>), publiées dans des livres, revues et conférences internationales, dont 3 prix de meilleur papier étudiant obtenus en conférence nationales et internationales.</p>
Enseignements & Encadrements	<p>Acteur de l'enseignement universitaire par des cours magistraux, travaux dirigés et encadrement de projets, depuis 2001.</p> <p><input type="checkbox"/> <i>Heures effectuées : plus de 700 heures (équivalent TD)</i></p> <p><input type="checkbox"/> <i>Etablissements : Ecole Centrale de Paris, Institut Henri Poincaré, ENSICAEN, Université de Caen, Université de Nice, Ecole Supérieure en Sciences Informatiques de Nice-Sophia Antipolis, ENIT Tunis.</i></p> <p><input type="checkbox"/> <i>Co-encadrement des doctorats de H-E. Assemblal (2007-2009) et M. Daisy (2012-2015) et du post-doctorat de P. Buysens (2013-2015).</i></p>
Développements logiciels	<p>Auteur de logiciels et bibliothèques libres : Ces projets sont très actifs, en développement permanent.</p> <p><input type="checkbox"/> <i>G'MIC, GREYC's Magic for Image Computing :</i> http://gmicro.eu (+de 150kloc, +de 3M téléchargements depuis 2008).</p> <p><input type="checkbox"/> <i>The CImg Library, C++ Template Image Processing Library :</i> http://cimg.eu (+ de 60kloc, depuis 1999).</p>
Autres	<p>Anglais : Lu, écrit, parlé (anglais courant). Notions d'Allemand. Titulaire du permis B.</p>

A.1 Vue d'ensemble de mon parcours de chercheur

De nos jours, la résolution et la dimensionnalité des signaux et des images numériques que l'on peut acquérir ont littéralement explosé. Les images couleurs, les images radars multi-fréquence, les images médicales IRM illustrent quelques exemples de données où cette propriété de multi-dimensionnalité doit être absolument prise en compte lorsque l'on cherche à les traiter algorithmiquement de manière efficace : il existe très souvent une corrélation forte entre les différents canaux mesurés (les différentes composantes vectorielles des signaux ou images), et cet a priori doit être intégré intelligemment lors de l'élaboration d'algorithmes de traitement, car ces données vectorielles s'avèrent bien plus riches en informations que les signaux scalaires usuels. C'est cette constatation simple qui a guidé une grande partie de mon travail de recherche.

Dès le début de ma thèse (en 1999), je me suis donc attaqué au problème assez général de la *régularisation* d'images multi-valuées et multi-dimensionnelles, en me basant sur le formalisme des *méthodes variationnelles et des EDP* (équations aux dérivées partielles), qui modélisent les images comme des fonctions multi-dimensionnelles continues. Ces techniques ont connu un gain croissant de popularité dans le domaine du traitement d'images, au début des années 90. Elles autorisent en effet le développement d'algorithmes génériques ayant des comportements très flexibles et localement adaptatifs (non-linéaires), et ont des performances maintenant reconnues, notamment en ce qui concerne la prise en compte des discontinuités dans les images, qui sont généralement porteuses d'informations essentielles (contours, structures, textures, etc.). Comprendre comment ces méthodes pouvaient s'exprimer dans le cadre général d'images *multi-valuées*, pouvant être contraintes ou non-contraintes, a été le moteur de mes premiers travaux de recherche. Ceux-ci ont d'abord débouché sur la proposition de solutions aux problèmes de *débruitage*, de *reconstruction (inpainting)* et d'*interpolation non-linéaire* d'images couleurs, puis plus tard, de *désentreplacement* d'images par intégration temporelle de vecteurs mouvements, ou encore d'*estimation* de modèles tensoriels de diffusion à partir de données IRM bruitées.

Depuis 2004, et mon intégration au CNRS, j'ai pu continuer dans ces voies, en proposant tout d'abord une nouvelle classe d'EDP de régularisation ayant des propriétés intéressantes de préservation de structures courbées, pour le problème du débruitage d'images, tout en étant l'encadrant principal d'une thèse dont le sujet portait sur l'estimation de profils locaux de diffusion non gaussiens en IRMd. Plus récemment, j'ai fait une incursion dans le monde de l'infographie (automatisée), en dérivant l'utilisation d'indicateurs géométriques locaux usuellement rencontrés dans les méthodes EDP pour proposer un système original de transformation automatique d'images en croquis/dessins, et d'autres filtres de transformation artistiques divers (transfert de couleurs entre deux images, et colorisation semi-supervisée de croquis de type BD par exemple). D'un point de vue global, ma recherche se base sur des formalismes mathématiques assez généraux, et son champ d'application s'est donc toujours voulu assez large. Quelques exemples d'applications concrètes de mes travaux comprennent, entre autres, des résolutions de problèmes de débruitage, d'interpolation et de reconstruction de données manquantes (inpainting), d'estimation, de régularisation et de visualisation d'images volumiques d'IRM du tenseur de diffusion, de suivi des fibres dans la matière blanche (tractographie) à partir des images IRM traitées, de traitement d'images stéréoscopiques pour le cinéma 3D, ou encore l'analyse de la géométrie et du mouvement de tuyaux flexibles dans des séquences d'images acquises par rayons X.

Aujourd'hui, on peut remarquer un certain essoufflement de l'intérêt théorique porté aux méthodes EDP dans la communauté du traitement d'images et ce, pour plusieurs raisons. D'une part, ces méthodes ont été très étudiées et sont globalement bien maîtrisées pour les problèmes

liés aux images qu'elles permettent de traiter. Pour ma part, ces méthodes ont été notamment bien généralisées pour leur utilisation sur des données images *multi-valuées* et *multi-dimensionnelles*. A présent, peu de formalismes EDP réellement novateurs semblent émerger en traitement d'images. D'autre part, la plupart des méthodes EDP ont une action très *locale* sur les données à traiter, alors que dans le même temps, on assiste aujourd'hui à un véritable "boom" des formalismes *non-locaux* et *multi-échelles* (méthodes basées patches, et réseaux de neurones convolutionnels). Les méthodes non-locales produisent des résultats remarquables, dans tous les sous-domaines du traitement d'images. J'ai pensé qu'il était possible de faire converger EDP et méthodes basées patches, et j'ai initié en 2008 un travail de recherche exploratoire sur l'unification possible de certaines méthodes non-locales et des EDP / formulations variationnelles, en proposant de travailler sur une représentation des images dans des espaces à grandes dimensions. Mais malgré des pistes théoriques prometteuses, l'implémentation pratique de ces méthodes hybrides s'avère rapidement très coûteuse, à la fois en utilisation mémoire et en temps de calcul, du fait de la grande dimensionnalité de l'espace des représentations des images. Depuis 2013, j'ai donc décidé d'orienter mes travaux de recherche vers un thème plus général et probablement plus porteur (*le traitement d'images pour l'art numérique*), non centré sur un formalisme mathématique unique des algorithmes que j'étudie et que je veux proposer.

De part ma formation initiale d'ingénieur en informatique, je me suis également toujours efforcé d'implémenter "proprement" l'ensemble des algorithmes développés lors de mes travaux et de les mettre à la disposition de la communauté du traitement d'images. Cette recherche systématique de ré-utilisabilité et de reproductibilité scientifique depuis plus de quinze ans, m'a amené à concevoir un ensemble cohérent de *logiciels* et de *bibliothèques*, dont certaines parties importantes sont aujourd'hui distribuées sous forme de *logiciels libres* et utilisées dans de nombreux pays, aussi bien par le grand public que par les scientifiques du domaine de l'imagerie. Cette mise à disposition participe d'une part à la valorisation et au rayonnement des travaux de l'équipe auprès d'un public international de spécialistes et de non spécialistes et, d'autre part, plus localement, à l'aide au développement d'algorithmes pour les nouveaux membres du laboratoire (stagiaires, doctorants et permanents), en proposant une base logicielle solide sur laquelle s'appuyer. Ces développements logiciels m'ont également amené à réfléchir à des aspects plus généraux de *génie logiciel du traitement d'images*. Ces aspects logiciels ont compté pour une part non négligeable de mon travail de recherche et je détaillerai donc également les points importants de mes efforts dans ce domaine. N'étant pas spécialiste du domaine de l'ingénierie logicielle, je n'ai malheureusement pas pu valoriser ce travail de développement sous forme de publications dans des conférences ou des articles de revue spécialisés. Depuis plusieurs années, je m'efforce néanmoins de présenter régulièrement ces contributions logicielles dans diverses conférences, congrès internationaux et médias en lignes ayant pour thème le logiciel libre et/ou le graphisme (*RMLL*, *FOSDEM*, *Libre Graphics Meeting*, . . .), qui n'ont pas (encore) de systèmes d'édition de *proceedings*. Je reste persuadé que le logiciel libre va jouer dans le futur un rôle croissant dans la valorisation de la diffusion des résultats scientifiques, de par la reproductibilité intrinsèque qu'il permet de réaliser, et je m'efforce donc d'essayer de valoriser mon travail dans ce domaine, et ceci par des voies moins directes que des publications au sens classique du terme.

Intégration dans l'équipe IMAGE du GREYC

Le *GREYC* (*Groupe de Recherche en Informatique, Image, Automatique et Instrumentation*) de Caen est un laboratoire commun CNRS, Université de Caen, ENSICAEN, qui regroupe tout le

potentiel Caennais dans le domaine des STIC. Il est organisé en 7 équipes qui couvrent ensemble un large spectre de la recherche dans le domaine des STIC.

Plus spécifiquement, l'équipe IMAGE du GREYC est aujourd'hui l'une des plus importantes du laboratoire (en nombre de permanents). Elle est composée de nombreux chercheurs d'origines variées (traitement du signal, mathématiques, informatique, intelligence artificielle). Cette variété de compétences constitue l'un des atouts de l'équipe et lui permet d'aborder l'analyse du contenu des images selon plusieurs points de vue ou paradigmes de modélisation et de concevoir des algorithmes originaux intégrant ces différents points de vue. L'équipe se focalise principalement sur les approches géométriques et statistiques du traitement et de l'analyse d'images. L'équipe est organisée autour de deux thèmes de recherche principaux :

Thème 1. *Variationnel, EDP et Statistiques pour le traitement des images* (Méthodes variationnelles, Problèmes inverses, Optimisation, EDP et signaux sur graphes, Nuages de points et 3D, Statistiques en haute dimension pour l'image, Photographie computationnelle et Morphologie mathématique multivariée).

Thème 2. *Reconnaissance de formes et recherche d'information dans les images/vidéos.* (Reconnaissance de formes, Graphes et chaînes, Apprentissage de métriques, Descripteurs et apprentissage pour l'image, et Ingénierie des connaissances pour la conception d'applications en image).

L'originalité de l'équipe tient par ailleurs à une forte activité de *développement de logiciels et de capitalisation des connaissances* pour le traitement et l'analyse d'images. La diversité scientifique des membres de l'équipe lui permet d'aborder la résolution de problèmes suivant des approches à la fois numériques et/ou symboliques.

Ma recherche se situe naturellement dans le premier thème *Variationnel, EDP et Statistiques pour le traitement des images*. Mon intégration dans l'équipe en Octobre 2004, s'est déroulée de manière très satisfaisante, puisque je m'intéressais à des formalismes mathématiques déjà connus par certains membres de l'équipe (méthodes variationnelles) tout en apportant des compétences complémentaires, notamment dans le domaine du traitement des images multivaluées, de l'utilisation des EDP, et de manière plus applicative, sur les aspects de régularisation d'images. J'ai également pu enrichir l'arsenal des logiciels/bibliothèques développés dans l'équipe, de par mon expérience de développement de logiciels libres acquise durant mes années de thèse/postdoc à l'INRIA.

Depuis le 1er Octobre 2017, je suis responsable de l'équipe IMAGE du GREYC, succédant à *Jalal Fadili*.

A.2 Enseignement, formation et diffusion de la culture scientifique

A.2.1 Encadrements

- **Encadrement de thèse** : J'ai été le co-encadrant de la thèse de *Maxime Daisy* effectuée au GREYC (financée par un contrat "Grand emprunt" 3DS), avec le Pr. *O. Lézoray*. Cette thèse a été soutenue le 2 Décembre 2015 (durée 3 ans et 3 mois).

Sujet : "*Inpainting basé motif d'images et de vidéos appliqué aux données stéréoscopiques avec carte de profondeur*". (<http://www.theses.fr/2015CAEN2059>). Après

une année d'ATER à l'IUT de St Lô, Maxime Daisy est aujourd'hui salarié de l'entreprise InnerSense à Toulouse, où il dirige une équipe de R&D en vision par ordinateur. Je suis co-auteur de 16 publications avec Maxime Daisy.

- **Encadrement de thèse** : J'ai été co-encadrant de la thèse de *Haz-Edine Assemlal* effectuée à l'Université de Caen (bourse ministère), avec le Pr. *L. Brun*. Cette thèse a été soutenue le 11 Janvier 2010 (durée : 3 ans et 3 mois).
Sujet : “*Etude des modèles de diffusion complexes pour l'IRM à haute résolution angulaire*” (<http://www.theses.fr/2010CAEN2001>). Après un *post-doctorat* de 2 ans à la prestigieuse Université *Mc Gill* à Montréal/Canada, *Haz-Edine Assemlal* est aujourd'hui salarié d'une entreprise canadienne spécialisée en imagerie IRM, issue de l'Université. Je suis co-auteur de 10 publications avec *Haz-Edine Assemlal*.
- **Encadrement de post-doctorat**. De Novembre 2013 à Septembre 2015, j'ai été co-encadrant, avec le Pr. *O. Lézoray* du post-doctorat de *P. Buysens*, effectué au GREYC (financé par un contrat “Grand emprunt” 3DS).
Sujet : “*Reconstruction de cartes de profondeurs de vues virtuelles stéréoscopiques resynthétisées*”. *Pierre Buysens* est aujourd'hui ingénieur R&D de l'entreprise *Cotral Lab* à Condé sur Noireau, spécialisé dans le design et la fabrication de protections auditives.
- **Aide au co-encadrement de thèse** : Dans une moindre mesure, j'ai participé au co-encadrement (durant trois mois) de la thèse de *M. El Hassani* (bourse CIFRE, NXP Semi-conducteurs), dirigée par *S. Jehan-Besson* et le Pr. *M. Revenu*, sur l’ “*Implémentation hardware d'algorithmes de segmentation d'images temps réel*”. Thèse soutenue le 9 Juin 2008 à l'Université de Caen.
- **Encadrement de stage de Master** : J'ai encadré, de Mars 2012 à Août 2012, un stage Master de 6 mois sur “*l'inpainting basé motifs à contraintes tensorielles*”, effectué par *Maxime Daisy*, recruté par la suite comme doctorant.
- **Encadrement de stage de Master** : J'ai encadré, de Mars à Août 2009, un stage Master de 6 mois sur “*l'inpainting rapide d'images basé motifs*”, effectué par *Steeve Macao*, en collaboration avec la start-up *Previznet*. A l'issue de ce stage, l'étudiant a été recruté comme ingénieur de recherche (en CDI) dans cette société.
- **Encadrement de stage de Master** : J'ai encadré, de Mars à Août 2006, un stage Master de 6 mois sur les “*Modèles de diffusion complexes pour l'IRM du tenseur de diffusion*”, effectué par *Haz-Edine Assemlal*, élève ingénieur de l'ENSICAEN, recruté par la suite comme doctorant.
- **Encadrement de stage de Master** : En 2005, j'ai encadré un stage Master de 6 mois sur le “*Suivi de mouvements de fils d'armures dans des séquences d'images de riser sous radioscopie X*”, effectué par *Jean Martinot*, élève ingénieur de l'ISIMA, en collaboration avec l'*Institut Français du Pétrole (IFP)*. A l'issue de ce stage, l'étudiant a été recruté à l'IFP pour un CDD d'un an.

A.2.2 Enseignement

- Le total de mes heures d’enseignements depuis 2001, s’évalue à plus de 700 heures équivalent TD.
- J’ai dispensé des cours magistraux et travaux dirigés sur les *EDP de diffusion* (niveau Master Recherche et 3ème année, environ 18-20 h.eq.TD annuels depuis 2011), et encadré de nombreux projets étudiants proposés à l’ ENSICAEN (*Ecole Nationale Supérieure d’Ingénieur de Caen*).
- J’ai dispensé des cours magistraux et travaux dirigés, encadré des projets étudiants depuis 2001, dans les établissements suivants : *Institut Henri Poincaré* (2008, Cours magistral, niveau Doctorat), *Ecole Centrale de Paris* (2005, Cours magistral + TDs), *Introduction au traitement d’image* (3 heures, Niveau : License L2, Université de Caen, 2005).
- J’ai participé en 2007, à une chaire UNESCO “*Mathematics for Life Sciences*” à Tunis/Tunisie, équivalent à la dispense de 4 jours de cours et TDs.
- Travaux Dirigés : *Traitement d’image en C++* (env. 60 heures). Niveau : IUT 2ème année, Université de Nice, 2001-2003.
- Mon support de cours principal est disponible en ligne : “Partial Differential Equations for Multi-Valued Image Regularization” https://tschumperle.users.greyc.fr/slides/slides_ensicaen2013.pdf.

A.2.3 Vulgarisation / Grand Public

- Entre 2012 et 2017, j’ai donné diverses présentations / démonstrations dans des conférences grand public, spécialisées dans le logiciel et le graphisme libre [T4], [T12], [T13], [T14], [T15], [T16], [T17].
- J’ai écrit une série d’articles de vulgarisation sur le traitement d’images dans la presse en ligne spécialisée logiciel libre : <http://linuxfr.org/tags/gmic/public>.
- Tous les ans, mon logiciel *G’MIC* est utilisé comme plateforme de démonstration sur le stand de l’équipe IMAGE du GREYC pour la “Fête de la Science”, et pour les visites de collégiens/lycéens venant au laboratoire.
- En 2012, j’ai été approché par une journaliste de “*Connexions*”, magazine de l’actualité de la recherche en basse normandie, pour l’écriture d’un article sur le logiciel *G’MIC* (paru).
- En 2006, 2008 et 2009, j’ai participé à l’animation d’un stand “*Traitement d’Images*” à la “Fête de la Science” à Caen, évènement annuel destiné à faire découvrir le monde scientifique au grand public.

- En 2007, j’ai tenu une conférence aux *Rencontres Mondiales du Logiciel Libre* (RMLL’2007) à Amiens, pour présenter la bibliothèque logicielle libre “*CImg*”, dont je suis l’auteur.
- En 2007, j’ai été interviewé dans le cadre d’un court reportage sur l’équipe IMAGE du GREYC, passé sur la chaîne locale du câble *CityzenTV* à l’occasion de la “*Fête de la Science*”. Vidéo disponible à l’adresse : http://www.dailymotion.com/leelool1995/video/x38b2k_equipe-image-du-greycensicaen_tech
- J’ai participé au stand *Documents électroniques* tenu par le GREYC au salon *RSI’2006* (Rendez-Vous des Solutions Internet et des Systèmes d’Informations), à Deauville, en 2006.

A.3 Transfert technologique, relations industrielles et valorisation

- **Projet “Grand Emprunt”** : J’ai été responsable scientifique pour le GREYC du “Projet Action 3DS”, projet de collaboration de 3 ans entre 2012-2015, regroupant 9 partenaires académiques et industriels : *Binocle, Device-ALAB, ENS Louis Lumière, E2V Semiconductors, GREYC, INRIA Grenoble, Université Paris VIII, Thalès-Angénieux, Université Pierre et Marie Curie, CNRS*. Budget alloué pour le GREYC : 220.000 euros sur 3 ans. Notre contribution a consisté en l’élaboration d’algorithmes d’*inpainting* efficaces pour la désocclusion de zones dans des vues stéréoscopiques virtuelles resynthétisées, pour le cinéma et la télévision 3d. Ce projet de recherche m’a permis de recruter et d’encadrer un doctorant (3 ans, entre 2012-2015) et un post-doctorant (2 ans, entre 2013-2015).
- **ANR CARTES** : J’ai participé entre 2009 et 2012 à l’ANR *CARTES* (durée 30 mois) : “Capture aisée et rapide de traces d’empreintes sur scènes”, en collaboration avec la Sagem, l’Université de Caen, la société ELDIM, l’IRCGSN (Institut de recherche criminelle de la gendarmerie nationale), l’INPS (Institut National de Police Scientifique) et la SDPTS (Sous-Direction de la Police Technique et Scientifique). Le projet *CARTES* avait pour ambition de concevoir un outil de capture, rapide et facile d’emploi, d’empreintes et de traces révélées sur scènes de crime. Je me suis chargé de la partie “débruitage” des données images à analyser.
- **Contrat industriel** : J’ai encadré entre Mars et Août 2009, un stage Master de 6 mois, dans le cadre d’un contrat de recherche, dont j’étais le responsable pour le GREYC, avec la société *Previznet*, située à Paris. Ce stage a porté sur “*l’inpainting rapide d’images basé motifs*” (montant estimé du contrat : env. 20 000 euros).
- **Contrat industriel** : Etude de faisabilité pour la détermination numérique de fils d’armures à partir de radioscopie X de flexibles. Contrat avec l’IFP (*Institut Français du Pétrole*). Responsables : *David Tschumperlé, Jalal Fadili, Marinette Revenu* (GREYC IMAGE). Dates : *Avril 2005 à Décembre 2005*. Budget : 10 277 euros.
- **Contrat industriel** : Suivi du mouvement des fils d’armures dans les séquences d’images du flexible sous radioscopie X. Contrat avec l’IFP (*Institut Français du Pétrole*). Respon-

sables : *David Tschumperlé, Jalal Fadili, Marinette Revenu*. (GREYC IMAGE). Dates : Avril 2006 à Décembre 2006. Budget : 16 996 euros.

- **Collaboration avec la douane** : Entre Septembre et Décembre 2015, j’ai collaboré avec le SNDJ (*Service National de Douane Judiciaire*) pour automatiser la suppression de labels indésirables sur des duplicatas de relevés bancaires scannés, afin d’améliorer la reconnaissance OCR et l’analyse automatique de mouvements bancaires potentiellement frauduleux. Via notre logiciel *G’MIC* intégré à *GIMP*, j’ai pu rapidement mettre en oeuvre un algorithme adapté et le distribuer pour le déploiement dans le service de douane judiciaire. Cet algorithme est entré en production pour les enquêteurs. Une lettre de remerciement de la directrice du service national de la douane judiciaire atteste de la réussite de cette collaboration.
- **Collaboration inter-laboratoire** : Equipe IMAGE du GREYC / Equipe GIN de CYCERON (CEA, CNRS, INSERM) de Caen, sur l’imagerie IRM de diffusion, dans le cadre de la thèse de H-E Assemblal dont j’ai été le co-encadrant. L’ensemble de l’équipe GIN de CYCERON ayant déménagé à Bordeaux, cette collaboration a dû s’arrêter.
- **Collaboration avec la start-up Laoviland**. J’ai co-encadré pendant l’année scolaire 2014-2015, deux projets 3A pour les étudiants de l’ENSICAEN, en collaboration avec la start-up LAOVILAND (<http://www.laoviland.com/>), sur les sujets suivants : 1. Création d’animations à partir d’images fixes. et 2. Fusion d’images multiples par détermination automatique des modes de blending.
- **Logiciels libres** : Je suis l’auteur principal et le responsable de logiciels/bibliothèques libres, mis à disposition du grand public : *CImg* (<http://cimg.eu/>) et *G’MIC* (<http://gmimc.eu/>). Ces logiciels participent à leur façon à la reconnaissance de l’équipe IMAGE du GREYC comme un acteur intéressant du logiciel libre en traitement d’images (plus de 3.000.000 de téléchargements comptabilisés à ce jour).

A.4 Encadrement, animation et management de la recherche

- **Rapporteur de thèse de doctorat** (thèse à l’étranger, ne nécessitant pas d’habilitation pour être rapporteur) :
 - Thèse de *Sune H. Keller* (Université de Copenhague/Danemark, sous la direction du Pr. Mads Nielsen et de François Lauze) : “*Video Upscaling Using Variational Methods*”, soutenue en Décembre 2007.
- **Membre de jury de thèse** (en tant qu’examinateur) :
 - Thèse de *Laszlo Marak* (Université Paris-Est, Laboratoire d’informatique Gaspard-Monge), sous la direction du Pr. Laurent Najman et d’Hugues Talbot, soutenue en Mars 2012. Titre : “*On continuous maximum flow image segmentation algorithm*”. <http://www.theses.fr/2012PEST1093>
 - Thèse de *Denis Fortun* (IRISA - Campus de Beaulieu - Rennes), sous la direction du Pr. Patrick Bouthémy et de Charles Kervrann, soutenue en Juillet 2014. Titre :

“Aggregation framework and patch-based representation for optical flow”. <http://www.theses.fr/2014REN1S093>

- **Membre nommé du CNU 27ème section** pour la période 2012-2015 (participation annuelle à 1/3 des sessions environ).
- **Membre du comité de pilotage du réseau Normandev**, entre Mars 2014 et Août 2015. Le réseau NormanDEV est affilié au réseau national du CNRS des acteurs du DEVeloppement LOGiciel au sein de l’Enseignement Supérieur et de la Recherche (*DEVLOG*) et représente les régions de Basse et Haute Normandie. Le réseau organise des manifestations régulières autour du développement logiciel.
Page web du réseau : <http://normandev.cnrs.fr/>
- **Membre du comité scientifique** des journées inter-régionales de formation en neuroimagerie (*JIRFNI’09*) qui ont eu lieu à l’automne 2009, à Caen.
- **Membre du comité de relecteurs** pour les conférences internationales *SSVM’2009* (Workshop on Scale Space and Variational Methods), *CVPR’2009* (IEEE Conference on Computer Vision and Pattern Recognition), *ECCV’2008* (European Conférence on Computer Vision), *ICCV’2007* (IEEE International Conference on Computer Vision), *MICCAI’2007* (International Conference on Medical Image Computing and Computer Assisted Intervention), et pour les conférences nationales *GRETSI’2017*, *ORASIS’2017* (Congrès des jeunes chercheurs en vision par ordinateur), *RFIA’2010* (Reconnaissance des Formes et Intelligence Artificielle), *ORASIS’2011*, *ORASIS’2010*, *ORASIS’2009* et *RFIA’2008*.
- **Relecteur régulier** pour les revues : *IEEE Transactions on Image Processing*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Biomedical Engineering*, *Pattern Recognition*, *SIAM Journal of Imaging Sciences*, *Journal of Mathematical Imaging and Vision*, *Journal of Medical Image Analysis*, *Journal of Electronic Imaging*, *Journal of Visual Communication and Image Representation*, . . .

Liste de mes publications, présentations et logiciels scientifiques

La plupart des publications listées ci-dessous sont accessibles librement sous forme de fichiers .pdf, sur ma page web professionnelle, à l'adresse :
<http://tschumperle.users.greyc.fr/publications.php>

B.1 Livres et ouvrages (2)

[B2] **D. Tschumperlé**. *G'MIC : The Handbook*. Reference manual of the G'MIC image processing framework, published under the GNU Free Documentation License. 419 pages.
(*This book is continuously updated, on each new release of the software.*)

[B1] **D. Tschumperlé**. *PDE-Based Regularization of Multivalued Images and Applications*. PhD Thesis from the University of Nice-Sophia Antipolis/France, December 2002, 193 pages.
(*Awarded 2nd of the Best 2002-2003 PhD Thesis in the "Telecom Valley"*, also one of the 13 PhD Thesis selected for the ERCIM's Cor Baayen Award 2004 contest).

B.2 Chapitres d'ouvrage (6)

[BC6] **D. Tschumperlé**. *Anisotropic Diffusion PDEs for Regularization of Multichannel Images : Formalisms and Applications*. in Numerical Color Imaging, Editors : C. Fernandez-Maloigne, F. Robert-Inacio, L. Macaire, Publisher : Wiley, John & Sons, Incorporated, 03/2012.

[BC5] **D. Tschumperlé**. *EDP de Diffusion Anisotrope pour la Régularisation d'Image Multi-Canaux : Formalismes et Applications*. in Imagerie couleur numérique : Avancées et Perspectives, Editors : C. Fernandez-Maloigne, L. Macaire, F. Robert-Inacio, Publisher : Hermes, 2011.

[BC4] **D. Tschumperlé**, L. Brun. *Non-Local Regularization and Registration of Multi-Valued Images by PDE's and Variational Methods on Higher Dimensional Spaces*. in Recent Advances in Mathematical Image Processing, Springer, 2011.

[BC3] **D. Tschumperlé**. *Anisotropic Diffusion PDE's for Image Regularization and Visualization*. in Handbook of Mathematical Methods in Imaging, 1st Ed., Springer, 2010, ISBN : 978-0-387-92920-0. <http://www.springer.com/mathematics/applications/book/978-0-387-92920-0>

[BC2] **D. Tschumperlé**, R. Deriche. *Anisotropic Diffusion Partial Differential Equations in Multi-Channel Image Processing : Framework and Applications*. Advances in Imaging and Electron Physics (AIEP), Academic Press, 2007, ISBN : 978-0-12-373907-0.

[BC1] R. Deriche, **D. Tschumperlé**, C. Lenglet, M. Rousson. *Variational Approaches to the Estimation, Regularization and Segmentation of Diffusion Tensor Images*. Mathematical Models of Computer Vision : The Handbook (Paragios, Chen & Faugeras Eds), Springer, 2005, ISBN : 0387263713.

B.3 Revues internationales à comité de lecture (10)

[J10] P. Buysens, M. Daisy, **D. Tschumperlé**, O. Lézoray. *Exemplar-Based Inpainting : Technical Review and New Heuristics for Better Geometric Reconstructions*. IEEE Transactions on Image Processing, Vol. 24, No. 6, pp 1809–1824, June 2015.
(Journal ranked *Q1* on *SJR*, h-index : 182)

[J9] H-E Assemlal, **D. Tschumperlé**, L. Brun, K. Siddiqi. *Recent Advances in Diffusion MRI Modeling : Angular and Radial Reconstruction*. Journal of Medical Image Analysis, Vol. 15, No. 4, pp. 369–396, August 2011.
(Journal ranked *Q1* on *SJR*, h-index : 83)

[J8] H-E Assemlal, **D. Tschumperlé**, L. Brun. *Efficient and Robust Computation of PDF Features from Diffusion MR Signal*. Journal of Medical Image Analysis, Vol. 13, No. 5, pp. 715–729, 2009, ISSN : 1361-8415.
(Journal ranked *Q1* on *SJR*, h-index : 83)

[J7] M. Elhassani, S. Jehan-Besson, L. Brun, M. Revenu, M. Duranton, **D. Tschumperlé**, D. Rivasseau. *A Time-Consistent Video Segmentation Algorithm designed for Real-Time Implementation*. Journal of VLSI Design, Vol. 2008,
(Journal ranked *Q4* on *SJR*, h-index : 16) 2008.

[J6] **D. Tschumperlé**. *Fast Anisotropic Smoothing of Multi-Valued Images using Curvature-Preserving PDE's*. International Journal of Computer Vision, Vol. 68, No. 1, pp. 65–82, ISSN : 0920-5691.
(Journal ranked *Q1* on *SJR*, h-index : 141)

[J5] **D. Tschumperlé**, R. Deriche. *Vector-Valued Image Regularization with PDE's : A Common Framework for Different Applications*. IEEE Transactions on Pattern Analysis and Machine

Intelligence, Vol. 27, No. 4, pp. 506–517, 2005.
(Journal ranked *Q1* on *SJR*, h-index : 241)

[J4] O. Faugeras, G. Adde, G. Charpiat, C. Chefd’hotel, M. Clerc, R. Deriche, G. Hermosillo, R. Keriven, P. Kornprobst, J. Kybic, C. Lenglet, L. Lopez-Perez, T. Papadopoulo, J-P. Pons, F. Segonne, B. Thirion, **D. Tschumperlé**, N. Wotawa. *Variational, Geometric and Statistical Methods for Modeling Brain Anatomy and Function*. NeuroImage, Vol. 23, No. 1, pp. 46–55, 2004
(Journal ranked *Q1* on *SJR*, h-index : 243)

[J3] C. Chefd’hotel, **D. Tschumperlé**, R. Deriche, O. Faugeras. *Manifold Constrained PDEs and Geometric Integration Methods in Image*. Journal of Mathematical Imaging and Vision, Vol. 20, No. 1-2, pp. 147–162, 2004, ISSN : 0924-9907.
(Journal ranked *Q1* on *SJR*, h-index : 48)

[J2] **D. Tschumperlé**, R. Deriche. *Orthonormal Vector Sets Regularization with PDE’s and Applications*. International Journal of Computer Vision, Vol. 50, No. 3, pp. 237–252, 2002, ISSN : 0920-5691.
(Journal ranked *Q1* on *SJR*, h-index : 141)

[J1] **D. Tschumperlé**, R. Deriche. *Diffusion PDE’s on Vector Valued Images, Local Approach and Geometric Viewpoint*. IEEE Signal Processing Magazine, Vol. 19, No. 5, pp. 16–25, 2002, ISSN : 1053-5888.
(Journal ranked *Q1* on *SJR*, h-index : 117)

B.4 Actes de colloques internationaux à comité de lecture (32)

[C32] M. Daisy, P. Buysens, **D. Tschumperlé**, O. Lezoray. *Exemplar-based Video Completion with Geometry-guided Space-time Patch Blending*. ACM SIGGRAPH Asia 2015, Kobe, November 2015.

[C31] P. Buysens, M. Daisy, **D. Tschumperlé**, O. Lezoray. *Depth-Aware Patch-based Image Disocclusion for Virtual View Synthesis*. ACM SIGGRAPH Asia 2015, Kobe, November 2015.

[C30] M. Daisy, P. Buysens, **D. Tschumperlé**, O. Lezoray. *Tensor-directed Spatial Patch Blending for Pattern-based Inpainting Methods*. International Conference on Computer Analysis of Images and Patterns (CAIP’15), Valletta/Malta, September 2015.

[C29] P. Buysens, M. Daisy, **D. Tschumperlé**, O. Lezoray. *Superpixel-based depth map inpainting for RGB-D view synthesis*. IEEE International Conference on Image Processing (ICIP’15), Québec City/Canada, September 2015.

[C28] M. Daisy, P. Buysens, **D. Tschumperlé**, O. Lezoray. *A Smarter Exemplar-based Inpainting Algorithm using Local and Global Heuristics for more Geometric Coherence*. IEEE International Conference on Image Processing (ICIP'14), Paris/France, October 2014.
(Selected in the top 10% papers of the conference)

[C27] M. Daisy, **D. Tschumperlé**, O. Lezoray. *A Fast Spatial Patch Blending Algorithm for Artefact Reduction in Pattern-based Image Inpainting*. ACM SIGGRAPH Asia 2013 Technical Briefs, Hong-Kong, November 2013.

[C26] M. Daisy, **D. Tschumperlé**, O. Lezoray. *Spatial patch blending for artefact reduction in pattern-based inpainting techniques*. International Conference on Computer Analysis of Images and Patterns (CAIP'13), York/UK, August 2013.

[C25] **D. Tschumperlé**. *The CImg Library*. IPOL 2012 Meeting on Image Processing Libraries, Cachan/France, June 2012.

[C24] **D. Tschumperlé**. *Tensor-Directed Simulation of Strokes for Image Stylization with Hatching and Contours*. IEEE International Conference on Image Processing (ICIP'11), Brussel/Belgium, September 2011.

[C23] H-E Assemlal, **D. Tschumperlé**, L. Brun. *Evaluation of q -Space Sampling Strategies for the Diffusion Magnetic Resonance Imaging*. International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI'09), London/England, September 2009.

[C22] **D. Tschumperlé**, L. Brun. *Non-Local Image Smoothing by Applying Anisotropic Diffusion PDE's in the Space of Patches*. IEEE International Conference on Image Processing (ICIP'09), Cairo/Egypt, November 2009.

[C21] H-E Assemlal, **D. Tschumperlé**, L. Brun. *Efficient Computation of PDF-Based Characteristics from Diffusion MR Signal*. International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI'08), New York/USA, September 2008.

[C20] H-E Assemlal, **D. Tschumperlé**, L. Brun. *Robust Variational Estimation of PDF functions from Diffusion MR Signal*. MICCAI Workshop on Computational Diffusion MRI (CDMRI'08), New York/USA, September 2008.

[C19] **D. Tschumperlé**, L. Brun. *Image Denoising and Registration by PDE's on the Space of Patches*. International Workshop on Local and Non-Local Approximation in Image Processing (LNLA'08), Lausanne/Switzerland, August 2008.

[C18] H-E. Assemlal, **D. Tschumperlé**, L. Brun. *Fiber Tracking on HARDI Data Using Robust ODF Fields*. IEEE International Conference on Image Processing (ICIP'07), Vol. 3, pp.

133-136, San Antonio/USA, September 2007, ISSN : 1522-4880.

[C17] **D. Tschumperlé**, Y. Bentolila, J. Martinot, J. Fadili. *Fast Time-Space Tracking of Smoothly Moving Fine Structures in Image Sequences*. IEEE International Conference on Image Processing (ICIP'07), Vol. 6, pp. 317–320, San Antonio/USA, September 2007, ISSN : 1522-4880.

[C16] M. Elhassani, S. Jehan-Besson, L. Brun, M. Revenu, M. Duranton, **D. Tschumperlé**, D. Rivasseau. *A Time-Consistent Video Segmentation Algorithm designed for Real-Time Implementation*. IEEE International Conference on Electronics, Circuits and Systems (ICECS'06), pp. 636–639, Nice/France, December 2006, ISBN : 1-4244-0395-2.

[C15] **D. Tschumperlé**, J. Fadili, Y. Bentolila. *Wire Structure Pattern Extraction and Tracking From X-Ray Images of Composites Mechanisms*. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'06), Vol. 2, pp. 2461–2466, New York/USA, June 2006.

[C14] M. El Hassani, S. Jehan-Besson, **D. Tschumperlé**, L. Brun, M. Revenu, D. Rivasseau, M. Duranton. *Vectorization of a Statistical Segmentation Algorithm*. International Congress of Imaging Science (ICIS'06), pp. 321–324, Rochester/USA, May 2006, ISSN : 0-89208-260-7.

[C13] **D. Tschumperlé**. *Curvature-Preserving Regularization of Multi-Valued Images using PDE's*. European Conference on Computer Vision (ECCV'06), LNCS Vol. 3952, pp. 295–307, Graz/Austria, May 2006, ISSN : 302-9743.

[C12] G. Forbin, B. Besserer, J. Boldys, **D. Tschumperlé**. *Temporal Extension to Exemplar-based Inpainting Applied to Scratch Correction in Damaged Images Sequences*. International Conference on Visualization, Imaging and Image Processing (VIIP'05), Benidorm/Spain, September 2005.

[C11] **D. Tschumperlé**. *LIC-Based Regularization of Multi-Valued Images*. IEEE International Conference on Image Processing (ICIP'05), Vol. 3, pp. 533–536, Genoa/Italy, September 2005, ISBN : 0-7803-9134-9.

[C10] **D. Tschumperlé**, B. Besserer. *High Quality Deinterlacing using Inpainting and Shutter-Model Directed Temporal Interpolation*. International Conference on Computer Vision and Graphics (ICCVG'04), pp. 301–307, Warsaw/Poland, September 2004, ISBN : 1402041780.

[C9] R. Deriche, **D. Tschumperlé**, C. Lenglet. *DT-MRI Estimation, Regularization and Fiber Tractography*. IEEE International Symposium on Biomedical Imaging : Nano to Macro (ISBI'04), Vol. 1, pp. 9–12, Arlington/USA, Avril 2004.

[C8] **D. Tschumperlé**, R. Deriche. *Tensor Field Visualization with PDE's and Application to DT-MRI Fiber Visualization*. IEEE Workshop on Variational and Level Set Methods (VLISM'03), Nice/France, October 2003.

[C7] **D. Tschumperlé**, R. Deriche. *Variational Frameworks for DT-MRI Estimation, Regularization and Visualization*. IEEE International Conference on Computer Vision (ICCV'03), Vol. 1, pp. 116–124, Nice/France, October 2003.

[C6] **D. Tschumperlé**, R. Deriche. *Vector-Valued Image Regularization with PDE's : A Common Framework for Different Applications*. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'03), Vol. 1, pp. 651–659, Madison/USA, June 2003.
(Awarded as the *Best Student Paper*)

[C5] **D. Tschumperlé**, R. Deriche. *DT-MRI Images : Estimation, Regularization and Applications*. International Conference on Computer Aided Systems Theory (EUROCAST'03), pp. 46–52, Las Palmas/Spain, January 2003.

[C4] C. Chedf'hotel, **D. Tschumperlé**, R. Deriche. *Constrained Flows of Matrix-Valued Functions : Application to Diffusion Tensor Regularization*. European Conference on Computer Vision (ECCV'02), LNCS Vol. 2350, pp. 251–265, Copenhagen/Denmark, June 2002, ISSN : 0302-9743

[C3] **D. Tschumperlé**, R. Deriche. *Diffusion Tensor Regularization with Constraints Preservation*. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'01), pp. 948–953, Kauai/Hawaii, December 2001.

[C2] **D. Tschumperlé**, R. Deriche. *Regularization of Orthonormal Vector Sets using Coupled PDE's*. IEEE Workshop on Variational and Level Sets Methods (VLISM'01), Vancouver/Canada, July 2001.
(Awarded as the *Best Student Paper*)

[C1] **D. Tschumperlé**, R. Deriche. *Constrained and Unconstrained PDE's for Vector Image Restoration*. Scandinavian Conference on Image Analysis (SCIA'01), Bergen/Norway, June 2001.

B.5 Actes de colloques nationaux à comité de lecture

(19)

[NC20] S. Fourey, **D. Tschumperlé**, D. Revoy. *Un algorithme semi-guidé performant de colorisation en aplats pour le dessin au trait*. Colloque sur le Traitement du Signal et des Images (GRETSI'17), Juan-les-Pins/France, September 2017.

[NC19] M. Daisy, P. Buysens, **D. Tschumperlé**, O. Lezoray. *Un Algorithme d'Inpainting par Motif Utilisant des Heuristiques Locales et Globales pour une Meilleure Cohérence Géométrique*. Colloque sur le Traitement du Signal et des Images (GRETSI'15), Lyon/France, September 2015.

[NC18] P. Buysens, M. Daisy, **D. Tschumperlé**, O. Lezoray. *Restauration de cartes de profondeur basée sur les superpixels pour la synthèse de vue RGB-D*. Colloque sur le Traitement du Signal et des Images (GRETSI'15), Lyon/France, September 2015.

[NC17] M. Daisy, P. Buysens, **D. Tschumperlé**, O. Lezoray. *Mélange Spatio-temporel de Patches pour une Réduction d'Artefacts Préservant la Géométrie des Images et des Vidéos*. Colloque sur le Traitement du Signal et des Images (GRETSI'15), Lyon/France, September 2015.

[NC16] M. Daisy, **D. Tschumperlé**, O. Lezoray. *Un algorithme rapide de fondu spatial pour la réduction d'artefacts visuels des méthodes d'inpainting "basés patch"*. Colloque sur la Compression et Représentation des Signaux Audiovisuels (CORESA'13), Le Creusot/France, November 2013.

(Awarded as the *Junior Researcher Award of the conference*)

[NC15] **D. Tschumperlé**, P. David, O. Lezoray. *Extension des pseudo-gris pour la visualisation d'images scalaires sur des périphériques d'affichage couleur à faible profondeur de bits*. Colloque sur le Traitement du Signal et des Images (GRETSI'13), Brest/France, September 2013.

[NC14] M. Daisy, **D. Tschumperlé**, O. Lezoray. *Fondu spatial de patches pour la réduction des artefacts dans des méthodes génériques d'inpainting basé motifs*. Colloque sur le Traitement du Signal et des Images (GRETSI'13), Brest/France, September 2013.

[NC13] **D. Tschumperlé**. *Stylisation d'image basée contours et hachures par simulation de tracés de traits à géométrie tensorielle*. Colloque sur le Traitement du Signal et des Images (GRETSI'11), Bordeaux/France, September 2011.

[NC12] H-E. Assemlal, **D. Tschumperlé**, L. Brun. *Evaluation de stratégies d'échantillonnages de l'espace Q pour l'imagerie par résonance magnétique de diffusion*. Reconnaissance des Formes et Intelligence Artificielle (RFIA'10), Caen/France, January 2010.

[NC11] H-E Assemlal, **D. Tschumperlé**, L. Brun. *Estimation de caractéristiques quelconques de la PDF à partir d'un signal IRM de diffusion*. Colloque sur le Traitement du Signal et des Images (GRETSI'09), Dijon/France, September 2009.

[NC10] **D. Tschumperlé**, L. Brun. *Lissage Non Local d'Images Multi-Valuées par Diffusion Anisotrope dans l'Espace des Patches*. Colloque sur le Traitement du Signal et des Images (GRETSI'09), Dijon/France, September 2009.

[NC9] H-E Assemlal, **D. Tschumperlé**, L. Brun. *Estimation variationnelle robuste de modèles complexes de diffusion en IRM à haute résolution angulaire et tractographie*. Colloque sur le Traitement du Signal et des Images (GRETSI'07), Troyes/France, September 2007.

[NC8] M. El Hassini, S. Jehan-Besson, D. Rivasseau, L. Brun, **D. Tschumperlé**, M. Revenu, M. Duranton. *Conception d'un algorithme robuste de segmentation vidéo pour des applications temps-réel*. Conférence sur la Compression et Représentation des Signaux Visuels (CORESA'06), Caen/France, November 2006.

[NC7] V. Do, G. Lebrun, L. Malapert, C. Smet, **D. Tschumperlé**. *Inpainting d'Images Couleurs par Lissage Anisotrope et Synthèse de Textures*. Reconnaissance des Formes et Intelligence Artificielle (RFIA'06), Tours/France, January 2006.

[NC6] **D. Tschumperlé**. *Régularisation d'Images Multi-valuées par Convolutions de Lignes Intégrales*. Traitement et Analyse de l'Information : Méthodes et Applications (TAIMA'05), Hammamet/Tunisie, September 2005.

[NC5] **D. Tschumperlé**, R. Deriche. *Formalismes Variationnels pour l'Estimation, la Régularisation et la Visualisation d'Images IRMd*. Reconnaissance des Formes et Intelligence Artificielle (RFIA'04), Toulouse/France, January 2004.

[NC4] **D. Tschumperlé**, R. Deriche. *Régularisation d'Images Multivaluées par EDP : Un Formalisme Commun pour Différentes Applications*. Traitement et Analyse de l'Information : Méthodes et Applications (TAIMA'03), Hammamet/Tunisie, October 2003.

[NC3] R. Deriche, **D. Tschumperlé**. *EDP, Images Multivaluées, Contraintes et Applications*. Le traitement d'images à l'aube du XXIème siècle, Paris/France, March 2002.

[NC2] **D. Tschumperlé**, R. Deriche. *Régularisation par EDP de Champs de Vecteurs Orthornormés et Applications*. Reconnaissance des Formes et Intelligence Artificielle (RFIA'02), Angers/France, January 2002.

[NC1] **D. Tschumperlé**, R. Deriche. *Restauration d'Images Vectorielles par EDP*. Reconnaissance des Formes et Intelligence Artificielle (RFIA'00), Paris/France, February 2000.

B.6 Logiciels (4)

[S4] **D. Tschumperlé**. *G'MIC : GREYC's Magic for Image Computing*. Open-source framework for generic image processing, released under the CeCILL v2.0 license since 2008. <http://gmic.eu/>. (development is still active. **reached more than 3.000.000 downloads**).

[S3] **D. Tschumperlé**. *The CImg Library : C++ Template Image Processing Library*. Open-source library for image processing, released under the CeCILL-C license since 1999, <http://cimg.eu/>. (development is still active).

[S2] **D. Tschumperlé**, J. Fadili. *FISPEX : Filamentary Structure Pattern Extraction*. Closed-source software for enhancement, extraction and tracking of fine moving structures in image sequences. Contrat cadre CNRS/IFP (Institut Français du Pétrole). (development has ended).

[S1] **D. Tschumperlé**. *GREYCstoration : Open source algorithms for image denoising and interpolation*. Open-source software for image regularization, under CeCILL v2.0 license, 2004–2008, <http://cimg.eu/greycstoration/>. (development has ended in 2009).

B.7 Conférences invitées dans des congrès (18)

Les présentations listées dans cette section ont rarement fait l'objet de publications dans des *proceedings*. Je le signale explicitement lorsque c'est le cas.

[T18] **D. Tschumperlé**, M. Daisy, P. Buysens, O. Lézoray. *Blending Methods and Other Improvements for Exemplar-based Image Inpainting Techniques* Journée Traitement d'Images INSA, Rouen, April 2015.

[T17] **D. Tschumperlé**, M. Daisy. *Le Traitement des Images Au Service de l'Art Numérique*. Journée Nationale de la Culture Numérique (JNCN 2014), Cherbourg, June 2014.

[T16] **D. Tschumperlé**, J. Boulanger, P. David. *A 2012-2013 retrospective of the G'MIC project : New features from artists/developers collaborations*. Libre Graphics Meeting (LGM'2014), Leipzig/Germany, April 2014.

[T15] **D. Tschumperlé**. *G'MIC (GREYC's Magic for Image Computing) : A full-featured framework for image processing with various interfaces*. Libre Graphics Meeting (LGM'2013), Madrid/Spain, April 2013.

[T14] **D. Tschumperlé**. *G'MIC (GREYC's Magic for Image Computing) : A full-featured image processing framework*. Free and Open source Software Developers' European Meeting (FOSDEM'2013), Brussels/Belgium, February 2013,

[T13] **D. Tschumperlé**. *G'MIC, un framework générique pour le traitement d'image*. Capitole du Libre, Toulouse/France, Novembre 2012.

[T12] **D. Tschumperlé.** *CImg et G'MIC, outils libres pour le traitement d'image*. 2ème Edition des Rencontres du Libre, Lion sur Mer, Janvier 2012.

[T11] **D. Tschumperlé.** *Variational methods and PDE's on the Space of Patches for Image Denoising and Registration*. Second conference "Mathematics and Image processing", Orléans/France, March 2010.
(in proceedings, Springer : <http://www.springer.com/us/book/9783642196034>).

[T10] **D. Tschumperlé.** *IRM de Diffusion : Algorithmes de traitement d'image pour l'estimation des caractéristiques locales de la diffusion*. JIRFNI'2009 : Journées Interrégionales de formation en Neuroimagerie, Caen/France, Octobre 2009.

[T9] **D. Tschumperlé.** *G'MIC and The CImg Library : Open-Source Toolboxes for the Processing of Generic Images*. "Matheron" Image Processing Simulation Tools Workshop, organized by Thalès. Palaiseau/France, June 2009.

[T8] **D. Tschumperlé.** *PDE's on the Space of Patches for Image Denoising and Registration*. Symposium on Patch-based Image Representation, Manifolds and Sparsity. Rennes/France, Avril 2009.

[T7] **D. Tschumperlé.** *Tensor-directed Smoothing of Multi-Valued Images with Curvature-Preserving Diffusion PDE's*. International Conference on Image and Signal Processing (ICISP'08), Cherbourg/France, July 2008.

[T6] **D. Tschumperlé.** *Processing of diffusion MRI datasets using variational tools and PDE's*. Journées Internationales de Metz "PDE and variational methods in life sciences" (JDM'08), Metz/France, April 2008.

[T5] **D. Tschumperlé.** *Processing of diffusion MRI datasets using variational tools and PDEs*. Workshop on Bioimaging II / PDE's of the Johann Radon Institute for Computational and Applied Mathematics (RICAM), Linz/Austria, November 2007.

[T4] **D. Tschumperlé.** *CImg : An open-source C++ library for image processing*. Rencontres Mondiales du Logiciel Libre (RMLL'2007), Amiens/France, July 2007.

[T3] **D. Tschumperlé.** *Fast Anisotropic Smoothing of Multi-Valued Images using Curvature-Preserving PDE's*. International Conference on Scientific Computation and Differential Equations (SciCADE'07), St Malo/France, July 2007.

[T2] **D. Tschumperlé.** *Fast Anisotropic Smoothing of Multi-Valued Images using Curvature-Preserving PDE's*. Journées Internationales de Metz "Mathématique et Images" (JDM'07), Metz/France,

May 2007.

[T1] **D. Tschumperlé**. *Fast Anisotropic Smoothing of Multi-Valued Images using Curvature-Preserving PDE's*. Mathematics and Image Analysis (MIA'06), Paris/France, September 2006.

B.8 Rapports de Recherche (7)

[R7] S. Fourey, **D. Tschumperlé**, D. Revoy. *Un algorithme semi-guidé performant de colorisation en aplats pour le dessin au trait*. Pre-print hal-01490269, Caen/France, April 2017.

[R6] S. Macao, **D. Tschumperlé**. *Un modèle tensoriel pour le guidage d'inpainting basé patch*. Research Report : "Les cahiers du GREYC", No. 13-01, Caen/France, January 2013.

[R5] **D. Tschumperlé**, L. Brun. *Defining Some Variational Methods on the Space of Patches : Application to Multi-Valued Image Denoising and Registration*. Research Report : "Les cahiers du GREYC", No. 08-01, Caen/France, February 2008.

[R4] H-E. Assemlal, **D. Tschumperlé**, L. Brun. *A Variational Framework for the Robust Estimation of ODFs From High Angular Resolution Diffusion Images*. Research Report : "Les cahiers du GREYC", No. 07-01, Caen/France, April 2007.

[R3] **D. Tschumperlé**. *Fast Anisotropic Smoothing of Multi-Valued Images using Curvature-Preserving PDE's*. Research Report : "Les cahiers du GREYC", No. 05-01, Caen/France, January 2005.

[R2] O. Faugeras, G. Adde, G. Charpiat, C. Chefd'hotel, M. Clerc, R. Deriche, G. Hermosillo, R. Keriven, P. Kornprobst, J. Kybic, C. Lenglet, L. Lopez-Perez, T. Papadopoulo, J-P. Pons, F. Segonne, B. Thirion, **D. Tschumperlé**, N. Wotawa. *Variational, Geometric and Statistical Methods for Modeling Brain Anatomy and Function*. INRIA Research Report No. RR-5202, Sophia-Antipolis/France, May 2004.

[R1] **D. Tschumperlé**, R. Deriche. *Vector-Valued Image Regularization with PDE's : A Common Framework for Different Applications*. INRIA Research Report No. RR-4657, Sophia-Antipolis/France, December 2002.

Fin du document