

Insertion de détail dans des figures autosimilaires

Houssam Hnaidi, Eric Guérin, Samir Akkouche

▶ **To cite this version:**

Houssam Hnaidi, Eric Guérin, Samir Akkouche. Insertion de détail dans des figures autosimilaires. 20ièmes Journées de l'AFIG 2007, Nov 2007, Marne la Vallée, France. hal-01537641

HAL Id: hal-01537641
https://hal.science/hal-01537641

Submitted on 12 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Journées de l'Association Francophone d'Informatique Graphique, Marne la Vallée, 2007

Insertion de détail dans des figures autosimilaires

Houssam Hnaidi, Eric Guérin et Samir Akkouche

LIRIS - Université Claude Bernard Lyon 1

Abstract

This paper deals with the approximation of natural objects using a fractal model equipped with a detail concept like wavelet transforms. The fractal model is a projected Iterated Function System (IFS) model. This model unifies the IFS model and a classical model used in computer graphics (free form representation with control points). The concept of wavelet transforms is used for adding a detail part to the fractal model. The approximation problem has a two steps formulation : a nonlinear fitting optimisation of the fractal model and a linear transform for the computation of the detail coefficients. We have used this approximation method to approximate two leaves border. Then, the detail concept has been used to transfer the geometric texture from one leaf to another, and also to amplify these geometric details.

Résumé

Cet article traite de l'approximation d'objets naturels grâce à un modèle fractal équipé d'un concept de détail comme celui présent dans les ondelettes. Le modèle fractal est un modèle IFS (Iterated Function System) projeté. Ce modèle unifie le modèle IFS et un modèle classique utilisés en infographie (déformation des objets grâce à des points de contrôle). Le concept des transformations par ondelettes est immité par l'ajout d'une partie de détail au modèle fractal. L'approximation se fait alors en deux phases : une phase d'optimisation non linéaire du modèle fractal et une phase de transformation linéaire qui permet de calculer les coefficients de détail. Nous avons employé cette méthode pour approximer la frontière de deux feuilles. Le concept de détail a ensuite été employé pour transférer la texture géométrique d'une feuille à une autre, ainsi que pour l'amplification de détails géométriques.

1. Introduction

La modélisation auto-similaire est un outil efficace pour produire des objets naturels qui possèdent d'emblée la propriété d'auto-similarité (arbres, montagnes, *etc.*). Les fractales, qui permettent de faire ce type de modélisation, ont été utilisées depuis longtemps pour synthétiser des plantes comme les fougères. Si les outils de synthèse sont très nombreux dans ce domaine, on ne peut pas en dire autant des outils d'analyse et d'approximation. Ces derniers doivent parvenir, à partir d'un objet naturel issu du monde réel, à extraire une représentation informatique qui le reproduise le plus fidèlement. Ces outils posent la problématique connue sous le terme de problème inverse. Jacquin a été l'un des pionniers dans la matière en introduisant une méthode de compression d'images basée sur un modèle fractal.

Dans un domaine connexe, les outils permettant une analyse multi-échelle des objets ont récemment connu une grande expansion, notamment par l'intermédiaire de ce qu'on l'on appelle les ondelettes. JPEG 2000 est un standard de compression d'image qui emploie cette théorie pour la compression des images.

A plusieurs reprises, des auteurs ont montré les liens qu'il pouvait y avoir entre la théorie des ondelettes et les fractales. Nous proposons dans ce papier de développer une méthode pour approximer les objets naturels basée sur un modèle fractal équipé d'un concept de détail comme celui présent dans les ondelettes. Notre modèle est capable de reconstruire ces objets de manière exacte et ce avec un paradigme multi-résolution. Nous avons appliqué ce modèle au transfert de textures géométriques de contours ainsi qu'à l'amplification de détails géométriques.

Dans une première partie, nous présentons un aperçu des travaux liés à cette étude. Dans une deuxième partie, nous introduisons quelques définitions pour la base théorique

de notre modèle. Notre contribution est introduite dans la troisième partie. Enfin, nous présentons quelques applications basées sur notre modèle.

2. Travaux liés

On peut classifier la géométrie fractale en deux domaines, les modèles stochastiques et les modèles déterministes. Le modèle stochastique d'un objet ou d'un phénomène est défini pour être un modèle où l'objet est représenté par un processus stochastique d'une ou plusieurs variables. Dans cette lignée, les travaux de Mandelbrot et Van Ness [Nes68] introduisent le concept de fBm (fractional Brownian motion). En revanche avec les modèles déterministes les phénomènes sont reproductibles, chaque instance de modèle génère une même figure. Parmi les modèles déterministes on peut citer les L-System [Lin68] qui ont plusieurs applications dans l'informatique graphiques [PH89, PL91, Smi84], ainsi que les IFS (Iterated Function System) [Bar88]. Fondés sur un formalisme mathématique rigoureux, les IFS sont un outil puissant pour l'analyse comme pour la synthèse d'objets fractals. Hutchinson [Hut81] fut le premier à les étudier en appliquant le théorème du point fixe aux ensembles de compacts d'un espace métrique complet. Barnsley [Bar93] a développé ce formalisme et l'a utilisé dans toute une série d'applications, entre autres, en infographie et en compression d'image.

Notre travail est basé sur une variante des IFS appelée IFS projeté [Zai98], ce modèle utilise le principe utilisé dans les formes à pôles comme Bézier par exemple. L'idée principale des IFS projetés est la séparation entre l'espace d'itération et l'espace de modélisation. Une forme à pôles consiste en deux parties: les points de contrôles et les fonctions de mélange. Les fonctions de mélange dans les IFS projetés deviennent des modèles IFS.

L'IFS projeté est efficace pour la modélisation des objets fractals, Zaïr [Zai98] est le premier à l'introduire et l'utilise pour la modélisation. Guérin [Gué02] a employé ce modèle pour l'approximation de courbes et de surfaces. Les avantages de ce modèle sont, premièrement qu'il s'agit de modèles simples à implémenter, et deuxièmement la séparation entre l'espace de modélisation qui est représenté par les points de contrôle, et l'espace d'itération qui est représenté par les IFS. L'utilisation des IFS projetés pour l'approximation de courbes et de surfaces pose le problème inverse qui est résolu par Guérin [GTB01] par la minimisation de la distance entre la courbe ou la surface originale et l'autre généré, mais cette méthode a besoin d'un modèle avec beaucoup de paramètres ce qui rend la méthode d'approximation complexe. Un autre inconvénient de l'IFS projeté est qu'il ne peut pas approximer tous les objets complexes.

Les modèles fractals possèdent la propriété d'autosimilarité. Cela signifie que l'objet est composé de parties qui lui ressemblent. La théorie des ondelettes [Mey87, Mal89] est utile pour étudier l'autosimilarité des signaux et des images, elle est utilisée pour la compression des images. Même si les ondelettes sont utiles et efficaces pour l'analyse et la synthèse des objets, la fonctionnelle utilisée (fonction d'ondelette et fonction d'échelle) dépend de l'application visée et non de l'objet lui-même.

L'autosimilarité est une propriété commune entre les IFS et les ondelettes, c'est pourquoi plusieurs personnes ont utilisés les IFS et les ondelettes ensembles pour analyser l'autosimilarité des objets [Dau88, Dau90, ET07].

3. La base théorique de notre modèle

Dans cette section nous expliquons la base théorique de notre modèle, en commençant par rappeler quelques définitions dans la théorie des IFS, puis dans les IFS projetés.

3.1. IFS

Ici on va mentionner les définitions principales dans la théorie des IFS.

Définition 1 (IFS) Soit $(\mathcal{X}, d)$ un espace métrique complet. Nous appelons IFS tout ensemble fini $\mathcal{T} = \{T_0, ..., T_{N-1}\}$ d'opérateurs contractants sur $\mathcal{X}$ [Bar88].

Définition 2 (théorème d'existence d'attracteur) Pour tout IFS $\mathcal{T}$, il existe un compact unique non vide A de $\mathcal{H}(\mathcal{X})$ tel que :

$$\begin{aligned} A &= \mathcal{T}A \\ &= T_0A \cup \ldots \cup T_{N-1}A \end{aligned}$$

A est appelé attracteur de $\mathcal{T}$ et sera noté $A(\mathcal{T})$.

Il est donc possible d'associer à tout IFS $\mathcal{T}$ un compact, appelé attracteur. Par définition, tout attracteur possède la propriété d'auto-similarité au sens large, c'est-à-dire :

$$A = T_0A \cup \ldots \cup T_{N-1}A$$

avec les T_i appartenant à une classe de fonctions contractantes.

Définition 3 (alphabet) Soit un IFS $\mathcal{T} = \{T_0, \ldots, T_{N-1}\}$.$\Sigma = \{0, ..., N-1\}$ sera appelé l'alphabet associé à $\mathcal{T}$.

En notant Σ^* l'ensemble des mots finis sur Σ, et Σ^w l'ensemble des mots infinis sur Σ, on a le résultat suivant (voir Barnsley [Bar88]) :

Soit $\theta = \theta_1\theta_2\theta_3 \ldots \in \Sigma^w$. Pour tout $\lambda \in \mathcal{X}$ la limite :

$$\lim_{j\to\infty} T_{\theta 1} \ldots T_{\theta j}\lambda$$

existe et est indépendante de λ.

On peut définir une fonction d'adressage, qui, à un mot infini sur Σ, associe un point de l'attracteur.

Définition 4 (fonction d'adressage) Soient $\mathcal{T}$ un IFS et Σ son alphabet associé. On appelle fonction d'adressage la fonction ϕ qui à mot infini de Σ fait correspondre un élément de $\mathcal{X}$ de la façon suivante :

$$\begin{aligned} \phi &: \Sigma^w \rightarrow \mathcal{X} \\ \theta &\mapsto \phi(\theta) = \lim_{j\to\infty} T_{\theta 1} \dots T_{\theta j} \lambda \end{aligned}$$

avec $\lambda \in \mathcal{X}$ quelconque.

La figure (1) donne un exemple d'IFS composé de quatre transformations affines qui sont chacune des combinaisons d'homothéties, translations et rotations.

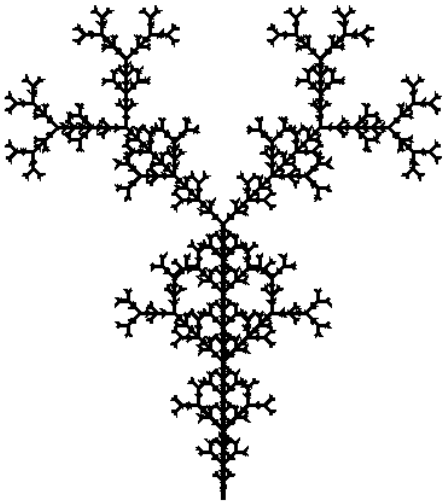


Figure 1: *Exemple d'IFS composé de transformations affines dans le plane*

3.2. IFS projeté

Pour fabriquer des fonctions de mélange fractales, il faut les définir en coordonnées barycentriques. Il faut donc choisir comme espace d'itération un espace barycentrique.

Définition 5 (espace barycentrique) Soit J un ensemble d'indices. L'espace barycentrique $\mathcal{B}^J$ associé à J est défini par :

$$\mathcal{B}^J = \left\{ (\lambda_j)_{j\in J} | \sum_{j\in J} \lambda_j = 1 \right\}$$

avec $\lambda_j \in \mathcal{R}$.

Il faut maintenant se fixer un semigroupe d'itération qui opère dans cet espace barycentrique. La solution la plus simple est d'utiliser des matrices dont les colonnes sont barycentriques.

Définition 6 (semigroupe de matrices barycentriques) Soit J un ensemble d'index. Le semigroupe de matrices barycentriques S_J associe à J est défini par :

$$S_J = \left\{ T | \sum_{j\in J} T_{ij} = 1, \forall i \in J \right\}$$

Grâce à ces choix, il est possible de ńprojeterż un attracteur d'IFS.

Définition 7 (attracteur d'IFS projeté) Soient J un ensemble d'indices, $\mathcal{T}$ un IFS constitué d'opérateurs de S_J, et $\mathcal{P} = (p_j)_{j\in J}$ une famille de points de contrôle. L'attracteur d'IFS projeté associé à $\mathcal{T}$ est défini par :

$$\begin{aligned} \mathcal{PA}(\mathcal{T}) &= \{\mathcal{P}\lambda | \lambda \in \mathcal{A}(\mathcal{T})\} \\ &= \mathcal{P}\phi(\theta) \\ &= \sum_{j\in J} \phi_j(\theta) p_j \end{aligned}$$

où $\mathcal{P}\lambda = \sum_{j\in J} \lambda_j p_j$

La figure (2) est un exemple de courbe fractale à pôle. Elle a été générée avec quatre points de contrôle et deux transformations.

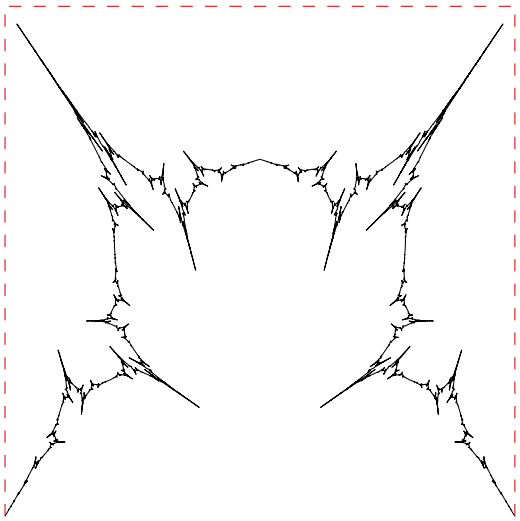


Figure 2: *Exemple de courbe généré avec IFS projeté*

3.3. Notre modèle

Afin de présenter une formule qui représente les détails insérés dans notre modèle fractals, on va commencer par présenter un moyen de visualiser les IFS projetés.

Définition 8 (visualisation d'attracteurs d'IFS projetés) Soient $(\mathcal{X}, d)$ un espace métrique complet, $\mathcal{T}$ un IFS, et $\mathcal{P}$ un vecteur de points de contrôle. Soit la suite définie par :

$$(\mathcal{S}_n)_{n\in\mathcal{N}} = \begin{cases} \mathcal{S}_0 & = \{\mathcal{P}\} \\ \mathcal{S}_{n+1} & = \mathcal{S}_n \mathcal{T}, \forall n \in \mathcal{N} \end{cases}$$

Alors, on a :

$$\lim_{n\to\infty} \mathcal{S}_n K = \mathcal{PA}(\mathcal{T})$$

avec $K \in \mathcal{H}(\mathcal{X})$.

$\mathcal{S}_n$ est un ensemble fini de polygones que l'on peut développer de manière récursive grâce à un arbre

$$\mathcal{S}_n = \mathcal{PT}^n = \left\{\mathcal{P}T_{\theta_1}\dots T_{\theta_n} \,|\, |\theta| = n\right\}$$

Afin de formaliser notre modèle on va écrire la suite précédente ainsi :

$$\mathcal{S}_n = \mathcal{P}T_\theta \quad \text{où} \quad T_\theta = T_{\theta_1}\dots T_{\theta_n} \quad \text{et} \quad |\theta| = n$$

Maintenant si on applique une seule transformation sur $\mathcal{S}_n$ alors on a :

$$\mathcal{S}_n T_i = \mathcal{P}T_\theta T_i \quad \text{où} \quad i \in \Sigma$$

en projetant le côté droit de l'équation précédente et en prenant la notation $\mathcal{P}T_\theta = \mathcal{P}_\theta$, on peut maintenant écrire :

$$\mathcal{P}_{\theta i} = \mathcal{P}_\theta T_i \qquad \text{où} \qquad i \in \Sigma$$

La figure (3) représente l'arbre utilisé pour visualiser cette formule. Cette figure montre le cas de deux transformations pour simplifier. On note P_ε le polygone initial de points de contrôle où ε représente le mot vide. Inspiré par les travaux

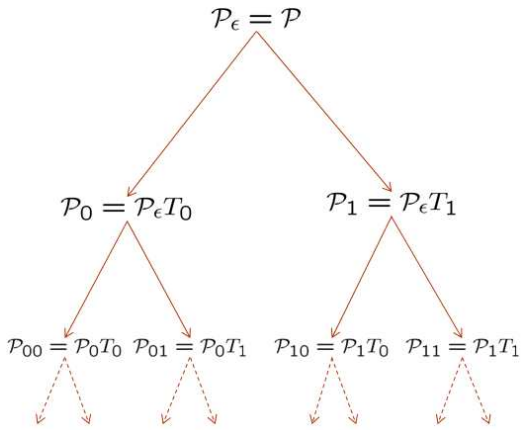


Figure 3: *Arbre de construction d'un IFS projeté*

de Tosan et al. [ET07] on peut ajouter une partie de détail sur la formule $\mathcal{P}_{\theta i} = \mathcal{P}_\theta T_i$ de cette manière :

$$\mathcal{P}_{\theta i} = \mathcal{P}_\theta T_i + \delta\mathcal{P}_\theta U_i \quad \text{où} \quad i \in \Sigma$$

Où $\delta\mathcal{P}_\theta$ est un vecteur de détail associé au polygone de points $\mathcal{P}_\theta$, et U_i (où $i \in \Sigma$) est une matrice que l'on appelle la matrice de déplacement de détail. La figure (4) représente l'arbre utilisé pour visualiser cette formule, aussi on utilise deux transformations et deux matrices de déplacement de détail pour simplifier le dessin. Comme on peut le remarquer sur la figure les détails sont ajoutés pour chaque niveau de l'arbre. Si on considère n points de contrôle, N transfor-

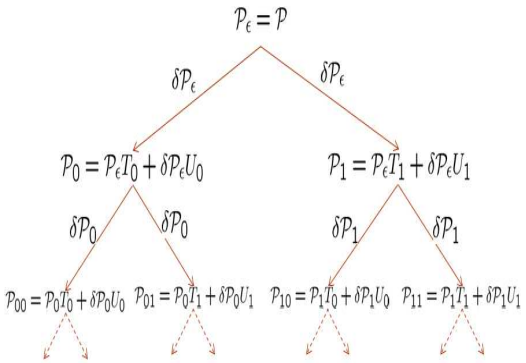


Figure 4: *Arbre de construction d'un IFS projeté équipé de détails*

mations, et on emploie l'espace $\mathcal{R}^k$, alors la dimension de la matrice T_i est $n \times n$, U_i est $n.(N-1) \times n$, $\mathcal{P}_\theta$ est $d \times n$, et $\delta\mathcal{P}_\theta$ est $d \times n.(N-1)$.

Le choix de la dimension des matrices de déplacement de détail et du vecteur de détail se base sur le fait que l'on cherche un système inversible.

En prenant la formule $\mathcal{P}_{\theta i} = \mathcal{P}_\theta T_i + \delta\mathcal{P}_\theta U_i$ pour tout $i \in \Sigma$ on a :

$$\begin{aligned} \mathcal{P}_{\theta 0} &= \mathcal{P}_\theta T_0 + \delta\mathcal{P}_\theta U_0 \\ \vdots \quad & \quad \vdots \quad \quad \vdots \\ \mathcal{P}_{\theta N-1} &= \mathcal{P}_\theta T_{N-1} + \delta\mathcal{P}_\theta U_{N-1} \end{aligned}$$

Maintenant on peut écrire les formules précédentes sous forme matricielle :

$$(\mathcal{P}_{\theta 0}|\dots|\mathcal{P}_{\theta N-1}) = (\mathcal{P}_\theta|\delta\mathcal{P}_\theta)\left(\begin{array}{c|c|c} T_0 & \dots & T_{N-1} \\ \hline U_0 & \dots & U_{N-1} \end{array}\right)$$

en utilisant la notation suivante :

$$R = \left(\begin{array}{c|c|c} T_0 & \dots & T_{N-1} \\ \hline U_0 & \dots & U_{N-1} \end{array}\right)$$

la formule devient :

$$(\mathcal{P}_{\theta 0}|\dots|\mathcal{P}_{\theta N-1}) = (\mathcal{P}_\theta|\delta\mathcal{P}_\theta)\,R$$

la formule dite inverse est

$$(\mathcal{P}_\theta|\delta\mathcal{P}_\theta) = (\mathcal{P}_{\theta 0}|\dots|\mathcal{P}_{\theta N-1})\,R^{-1}$$

ainsi R peut être apparenté à un filtre de synthèse et R^{-1} à un filtre d'analyse utilisés dans la transformée en ondelette.

4. Application de notre modèle

Un domaine où l'on peut utiliser notre modèle est la reconstruction d'objets. Le but est de pouvoir reconstruire n'importe quel objet avec un minimum d'information de détails. Pour cela, une phase préalable d'optimisation de la

matrice R est nécessaire. Dans la suite, nous expliquons la méthode utilisée pour cette optimisation dans le cadre de la reconstruction de courbes.

4.1. Méthode d'optimisation pour les courbes

Dans cette partie, nous considérons qu'une courbe est un ensemble ordonné de points. La méthode d'optimisation est basée sur la minimisation de la distance entre la courbe originale et la courbe reconstruite avec notre modèle en prenant comme paramètres la position des points de contrôle et la matrice R. On note les points de la courbe originale par p_i où $i = 0 \dots n-1$ et n est le nombre de points initiaux. On note les points générés avec notre modèle par P'_i. Il est important de noter ici que notre modèle permettant de reconstruire parfaitement les données d'entrée, nous allons volontairement omettre une partie de l'information de détail lors de la reconstruction. La distance entre chaque doublon de points est définie par :

$$d_i = \left\| p_i - p'_i \right\| \quad \text{où} \quad i = 0 \dots n-1$$

Nous avons choisi la méthode de Levenberg-Marquardt [PFTV93] pour minimiser cette distance. Il s'agit d'une méthode de régression non linéaire basée sur la différentiabilité de la fonction à minimiser. Formellement, elle s'exprime de cette manière, soient :

- n points dans le plan $(x_i, y_i)_{i=0\dots n-1}$
- une fonction $y = f(x,a)$, où a représente une vecteur des paramètres.

Le but est de trouver le vecteur de paramètres a_{opt} qui approxime le mieux les données c'est-à-dire :

$$f(x_i, a_{\text{opt}}) \approx y_i)_{i=0\dots n-1}$$

Dans notre cas, chaque donnée d'entrée (x_i, y_i) va correspondre au couple $(i,0)$ et la fonction f sera définie par $f(i,a) = d_i$ où a est le vecteur de paramètres représentant dans notre cas la position des points de contrôle et les valeurs de la matrice R.

La fonction de mérite $\mathcal{X}^2(a)$ utilisée dans la méthode de Levenberg-Marquardt est la suivante :

$$\mathcal{X}^2(a) = \sum_{i=0}^{n-1} (0 - f(i,a))^2 = \sum_{i=0}^{n-1} (0 - d_i)^2$$

La méthode étant différentielle, nous devons calculer numériquement la dérivée de f par rapport aux paramètres a_k :

$$\frac{\partial f(i,a)}{\partial a_k} = \frac{f(i,a)_{a_k+\varepsilon} - f(i,a)}{\varepsilon}$$

avec $k = 0 \dots m-1$ où m est le nombre de paramètres.

Nous avons utilisé la notation $f(i,a)_{a_k+\varepsilon}$ pour dire que la valeur a_k du vecteur de paramètres a été modifiée de ε.

La minimisation s'effectue en deux étapes :

- Dans la première étape, on minimise la distance entre la courbe originale et la courbe construite avec notre modèle en utilisant les positions de points de contrôle et les coefficient des matrices $\mathcal{T}$ comme paramètres. Dans ce cas, aucune information de détail n'est utilisée pour reconstruire la courbe.
- Dans la deuxième étape, on minimise cette distance en prenant les coefficient des matrices $\mathcal{U}$ comme paramètres. Dans ce cas, une partie de l'information de détail seulement est utilisée, sinon la reconstruction serait parfaite et il n'y aurait rien à minimiser. En pratique, les trois derniers niveaux de détails sont omis.

Les détails utilisés dans l'optimisation sont produit par application de la formule d'analyse $(\mathcal{P}_\theta | \delta\mathcal{P}_\theta) = (\mathcal{P}_{\theta 0} | \dots | \mathcal{P}_{\theta N-1}) R^{-1}$ en commençant par les points initiaux jusqu'à arriver aux points de contrôle.

Les Figures (5,6) représentent deux exemples d'utilisation de notre modèle pour reconstruire deux courbes. La première représente la frontière d'un feuille de pomme, et la deuxième représente la frontière d'un feuille de cerise.

Pour les deux exemples, notre modèle est composé de quatre points de contrôle, deux transformations et deux matrices de détails. Aussi dans les deux exemples on a reconstruit les courbes en omettant les derniers trois niveaux de détails. Cela signifie que l'on utilise un huitième de l'information initiale.

4.2. Transfert de texture géométrique

Nous pouvons appliquer notre modèle au transfert de texture géométrique en utilisant la possibilité d'extraction de détails. Les coefficients de détails (à divers niveaux) peuvent être déplacés d'un objet à un autre. Sur un même objets, il peuvent être amplifiés, diminués. La puissance et la flexibilité de notre modèle pour l'analyse et la reconstruction nous donnent la liberté de l'utiliser pour diverses applications. Par exemple on optimise notre modèle avec un objet, puis on applique le modèle optimisé sur un autre objet, après on modifie les détails du premier objet en utilisant les détails du deuxième objet. La possibilité inverse est offerte : on optimise notre modèle avec le deuxième objet puis on modifie aussi les détails du premier objet.

Le figure (7) est un exemple de transfert de texture entre deux objets. le modèle qui est optimisé avec la courbe originale (courbe (a)) est utilisé pour l'opération de transfert de texture. En revanche la figure (8) utilise le modèle qui est optimisé avec la courbe qui contient la texture pour la même opération.

Le figure (9) donne un exemple d'amplification et diminution de détails et leur influence sur la texture.

5. Conclusion

Nous avons présenté une méthode pour approximer les courbes basée sur un modèle fractal équipé d'un concept de détail comme celui utilisé dans les transformations en ondelettes. Nous avons utilisé ce modèle pour approximer les courbes et pour le transfert et l'amplification de texture géométriques. L'avantage le plus important de notre modèle est qu'il est simple, car il utilise un nombre de paramètre plus faible pour approximer les courbes qu'avec un modèle fractal normal non doté de détail. Nos envisageons de développer notre modèle afin de traiter les surfaces de profondeur ainsi que les surfaces générales.

References

[Bar88] BARNSLEY M.: *Fractals everywhere*. Academic Press, 1988.

[Bar93] BARNSLEY M.: *Fractals everywhere (second edition)*. Academic Press Professional, 1993.

[Dau88] DAUBECHIES I.: Time-frequency localization operators: A geometric phase space approach. *IEEE Transactions on Information Theory 34*, 4 (1988), 605–612.

[Dau90] DAUBECHIES I.: The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory 36*, 5 (1990), 961–1005.

[ET07] E. TOSAN I. BAILLY-SALINS I. S. G. G. Y. W.: Modelisation iterative de courbes et surfaces : aspect multiresolution. In *Groupe de travail en Modelisation Geometrique journee de Valenciennes* (mars 2007), pp. 55–69.

[GTB01] GUÉRIN E., TOSAN E., BASKURT A.: Approximation de courbes et surfaces par une méthode fractale. In *CORESA 2001* (Dijon, nov 2001).

[Gué02] GUÉRIN E.: *Approximation fractale de courbes et de surfaces*. Thèse de doctorat, Université Claude Bernard Lyon 1, dec 2002.

[Hut81] HUTCHINSON: Fractals and self-similarity. *Indiana University Journal of Mathematics 30* (1981), 713–747.

[Lin68] LINDENMAYER A.: Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs. *Journal of Theoretical Biology 18*, 3 (March 1968), 300–315.

[Mal89] MALLAT S.: A theory for multiresolution signal decomposition: The waveletrepresentation. *IEEE Trans. Pattern Anal. Mach. Intell. 11*, 7 (1989), 674–693.

[Mey87] MEYER Y.: Les ondelettes. In *Contributions to nonlinear partial differential equations, Vol. II (Paris, 1985)*, vol. 155 of *Pitman Res. Notes Math. Ser.* Longman Sci. Tech., Harlow, 1987, pp. 158–171.

[Nes68] NESS. B. M. J. V.: Fractal brownian motions, fractal noises, and applications. *SIAM Rewiew*, 10 (1968), 422–437.

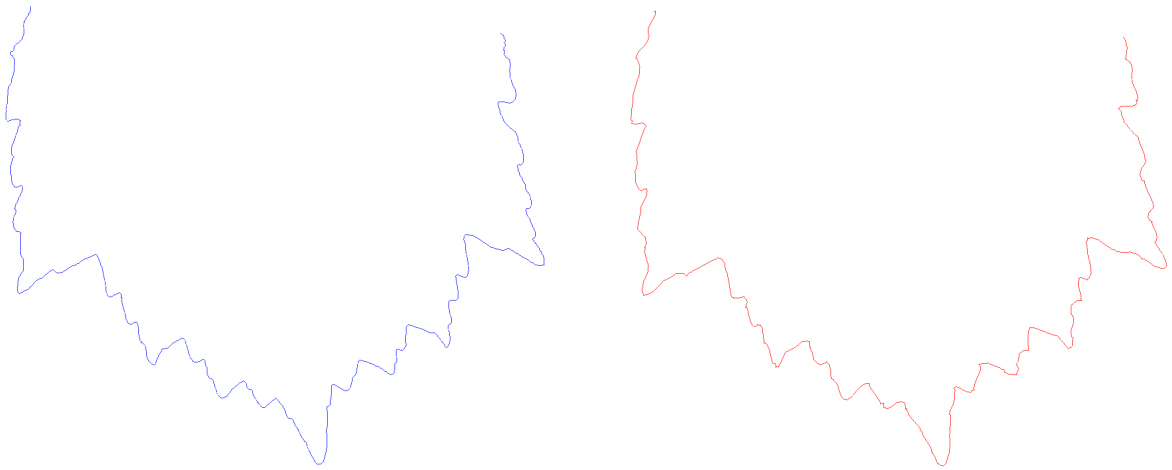
[PFTV93] PRESS W. H., FLANNERY B. P., TEUKOLSKY S. A., VETTERLING W. T.: *Numerical Recipes in C : The Art of Scientific Computing*. Cambridge University Press, 1993, ch. Nonlinear Models.

[PH89] PRUSINKIEWICZ P., HANAN J.: Lindenmayer systems, fractals, and plants. *Lecture Notes in Biomathematics 75* (1989).

[PL91] PRUSINKIEWICZ P., LINDENMAYER A.: *The Algorithmic Beauty of Plants (The Virtual Laboratory)*. Springer, October 1991.

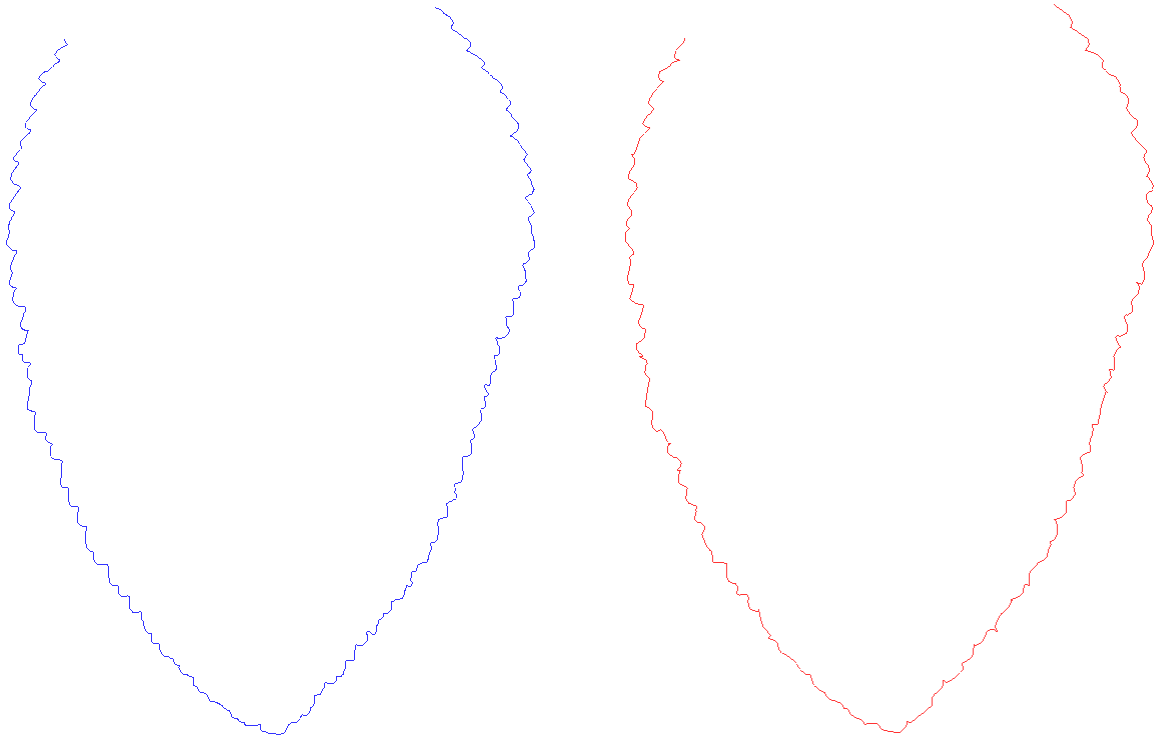
[Smi84] SMITH A. R.: Plants, fractals, and formal languages. In *SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques* (July 1984), vol. 18, ACM Press, pp. 1–10.

[Zai98] ZAIR C. E.: *Formes fractales à pôles basées sur une généralisation des IFS*. Thèse de doctorat, Université Claude Bernard Lyon 1, jun 1998.



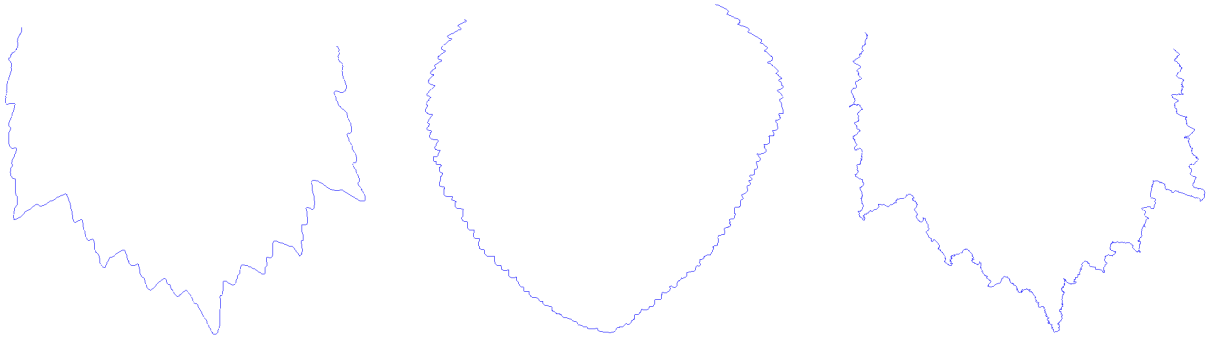
(a) (b)

Figure 5: *(a) La courbe originale de la feuille de pomme. (b) La courbe reconstruite en utilisant un huitième de l'information de la courbe originale.*



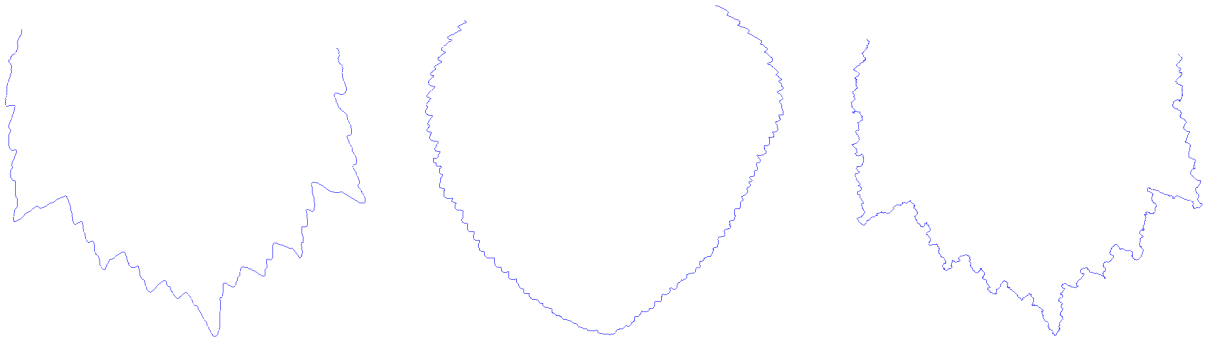
(a) (b)

Figure 6: *(a) La courbe originale de la feuille de cerise. (b) La courbe reconstruite en utilisant un huitième de l'information de la courbe originale.*



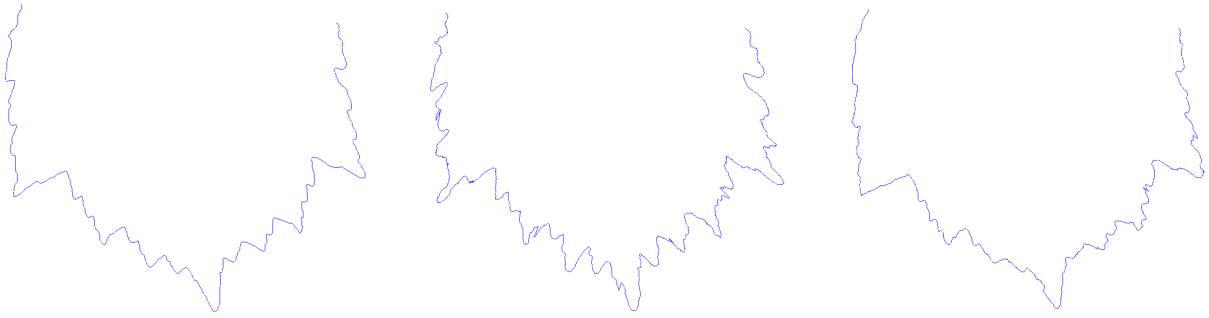
(a) (b) (c)

Figure 7: *Transfert la texture en utilisant le modèle de la courbe originale. (a) La courbe originale. (b) La courbe qui contient la texture. (c) La courbe originale avec la texture de la deuxième courbe.*



(a) (b) (c)

Figure 8: *Transfert la texture en utilisant le modèle de la courbe qui contient la texture. (a) La courbe originale. (b) La courbe qui contient la texture. (c) La courbe originale avec la texture de la deuxième courbe.*



(a) (b) (c)

Figure 9: *(a) La courbe originale. (b) La courbe avec texture doublée. (c) La courbe avec texture diminuée.*