

N° d'ordre : 76-2003

UNIVERSITE DE LIMOGES

Faculté des Sciences et Techniques

Ecole Doctorale Sciences-Technologie-Santé

Laboratoire MSI

THESE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE DE LIMOGES

Discipline : Informatique

Présentée et soutenue publiquement par

Karl MARITAUD

le 20 décembre 2003

Rendu réaliste d'arbres vus de près en images de synthèse

Directeur de Thèse : Djamchid GHAZANFARPOUR

Codirecteur : Jean-Michel DISCHLER

Jury

Rapporteurs : M. René CAUBET, professeur à l'Université de Toulouse

M. Pascal LIENHARDT, professeur à l'Université de Poitiers

Examineurs : M. Jean-Michel DISCHLER, professeur à l'université de Strasbourg

M. Djamchid GHAZANFARPOUR, professeur à l'Université de Limoges

M. Dimitri PLEMENOS, professeur à l'Université de Limoges

Remerciements

Ce travail a été réalisé au sein du Laboratoire « Méthodes et Structures Informatiques » (MSI) et a pu être mené à bien grâce au soutien financier de la région Limousin, sous la forme d'une bourse régionale sur deux ans. Il a été dirigé par les Professeurs Djamchid Ghazanfarpour et Jean-Michel Dischler que je remercie pour leur patience et la confiance qu'ils m'ont accordée.

Un grand merci également à messieurs René Caubet et Pascal Lienhardt qui ont accepté d'examiner mes travaux malgré leur emploi du temps très chargé et qui ont également bien voulu faire preuve de patience et de compréhension.

Je tiens à remercier chaleureusement l'ensemble des membres du Laboratoire MSI pour leur sympathie et plus particulièrement ceux que j'ai côtoyés quotidiennement, à commencer par Sébastien, Damien, Yann et Romain, qui ont partagé mon bureau et donc mes humeurs, bonnes et moins bonnes.

De même, je salue tous ceux que j'ai rencontrés à l'ENSIL puis à Jidé : enseignants, personnels administratif et technique, avec une mention spéciale pour Suzie, ainsi que les étudiants.

J'ai également une pensée toute particulière pour mes amis Guillaume, Astrid et Gaël qui m'ont soutenu au cours de cette thèse. Et, au risque de provoquer quelque moquerie de la part de ceux qui n'ont pas d'animaux domestiques et qui ne peuvent donc probablement pas me comprendre, je remercie mes adorables chats, Vanille et Titeuf, dont les ronrons ont un formidable pouvoir anti-stress... même si leurs miaulements ont parfois plutôt l'effet inverse.

Mais ces remerciements seraient bien évidemment incomplets sans l'immense reconnaissance que j'ai pour mes parents qui ont toujours fait passer mes études en premier et sans qui cette thèse n'aurait jamais pu voir le jour.

Résumé

« Rendu réaliste d'arbres vus de près en images de synthèse »

La plupart des travaux pour représenter des arbres en Infographie se sont orientés sur des méthodes conçues pour des points de vue éloignés où les détails de l'écorce et des feuilles ne sont pas perceptibles. Le rendu réaliste d'arbres vus de près reste un problème ouvert et difficile, essentiellement à cause de la complexité visuelle de l'écorce. Ici, des méthodes basées-image sont proposées pour extraire le relief et la couleur d'une texture réelle d'écorce à partir d'une simple photographie, sans manipulation compliquée, et ensuite pour synthétiser des textures 3D d'écorces réalistes. Une méthode est proposée pour appliquer ces textures, sans répétition ni discontinuité, sur des branches modélisées par déplacement de surfaces implicites mélangées sans gonflement. D'autre part, une technique est proposée pour extraire le relief des nervures des feuilles. Ensemble, ces méthodes nous permettent d'obtenir un haut niveau de réalisme visuel pour représenter des arbres vus de près.

Mots-clés : arbre, écorce, feuille, textures 3D, analyse et synthèse de texture, surface implicite, déplacement de surface (« *displacement mapping* »), mélange de volumes (« *blending* »), perturbation de normale (« *bump mapping* »).

Abstract

“Realistic rendering of trees viewed from a short distance in Image Synthesis”

Most of the research in the rendering of trees in Computer Graphics has focused on methods designed for viewpoints at a certain distance where the details of the bark and leaves are not perceptible. Rendering realistic trees at a short distance remains an open and difficult problem, essentially due to the visual complexity of bark textures. Here, image-based methods are proposed to extract the relief and color from a single photograph of real bark texture, without any complicated manipulations, and then to automatically synthesize realistic 3D bark textures. A method is proposed to apply these textures, with neither repetition nor discontinuities on the texture, onto branches modeled by displacement-mapped implicit surfaces, which are blended without bulge. Moreover, a technique for extracting the relief of leaf veins is proposed. Together, these methods allow us to achieve a high degree of visual realism for rendering close-up views of trees.

Key words: tree, bark, leaf, 3D textures, texture analysis and synthesis, implicit surface, displacement mapping, blending, bump mapping.

Table des matières

Introduction	21
Chapitre 1 Etat de l'art sur la représentation d'arbres en images de synthèse.....	31
1.1 Introduction	31
1.2 Notions de biologie	32
1.2.1 Qu'est-ce qu'un arbre ?	32
1.2.1.1 Terminologie	32
1.2.1.2 Classification	33
1.2.1.3 Arbre ou arbuste ?	34
1.2.2 La croissance des arbres	35
1.2.2.1 La forme de l'arbre en fonction de son environnement	35
1.2.2.2 Les branches	36
1.2.3 L'écorce	37
1.2.3.1 La structure de l'écorce	38
1.2.3.2 Différencier visuellement les différents types d'écorces	39
1.2.4 Les feuilles	41
1.2.4.1 Morphologie	41
1.2.4.2 Aspect	42
1.2.5 Les autres éléments d'un arbre (fruits, fleurs et racines)	43
1.3 Représentation d'arbres en infographie.....	44
1.3.1 Représentation « impressionniste »	45
1.3.1.1 Représentation par « panneaux » (images/photos 2D).....	45
1.3.1.2 La solution actuelle retenue en général dans les applications temps-réel ...	47
1.3.1.3 Systèmes de particules	48
1.3.1.4 Autres méthodes	50
1.3.2 Modélisation « structurelle » (du squelette de l'arbre)	50
1.3.2.1 Les débuts	51
1.3.2.2 Les modèles à base de « L-systèmes »	51
1.3.2.3 Un modèle combinatoire	53

1.3.2.4	Un modèle basé sur la géométrie	54
1.3.2.5	Les modèles basés sur des connaissances botaniques	55
1.3.2.6	Un modèle interactif.....	56
1.3.3	Du volume pour les branches	57
1.3.3.1	Cylindres	58
1.3.3.2	Troncs de cônes	59
1.3.3.3	Cône-sphères	60
1.3.3.4	Cylindres généralisés.....	60
1.3.3.5	Surfaces implicites	62
1.4	L'écorce.....	65
1.4.1	Placage 2D.....	65
1.4.1.1	Application de texture 2D sur des formes cylindriques	66
1.4.1.2	Application de texture sur des surfaces quelconques.....	67
1.4.2	Perturbation de normale (« <i>Bump mapping</i> »)	67
1.4.3	Carte de déplacement de surface (« <i>Displacement mapping</i> »)	68
1.4.4	Surfaces implicites perturbées.....	68
1.4.5	Texels	70
1.4.6	Synthèse des craquelures par système de masses-ressorts	70
1.5	Les feuilles	71
1.5.1	Modélisation (synthèse) des feuilles	72
1.5.2	Acquisition par photographies	73
1.5.3	Travaux sur la translucidité et la réflectance.....	74
1.6	Conclusion.....	75
Chapitre 2	Analyse et synthèse d'écorce	77
2.1	Introduction	77
2.2	Extraction du relief à partir d'une photographie	80
2.2.1	Corrélation entre les niveaux de gris et le relief.....	81
2.2.2	Retrouver le relief à partir de l'illumination : « <i>Shape from Shading</i> ».....	84
2.2.3	Extraction du relief par traitements sur les niveaux de gris de l'image	85
2.2.3.1	Le modèle 2D	86
2.2.3.2	Principe de la méthode	88
2.2.3.3	Classification.....	95

2.2.4	Extraction d'un relief cohérent par segmentation d'une image	99
2.2.4.1	Conditions à respecter pour le modèle	99
2.2.4.2	La macro-texture	100
2.2.4.3	La micro-texture	106
2.2.4.4	Mélange des deux couches de relief	112
2.2.4.5	Résultats	112
2.2.5	Une méthode combinant les deux précédentes	114
2.2.5.1	Retrouver la macro-texture	114
2.2.5.2	Reconstruction des niveaux intermédiaires de la macro-texture	116
2.2.5.3	Ajout de détails	119
2.2.5.4	Classification	119
2.2.6	Synthèse des trois méthodes proposées	122
2.3	Synthèse de texture d'écorce	123
2.3.1	Synthèse d'un ensemble d'échantillons de texture juxtaposables	125
2.3.1.1	Copie de morceaux de texture	127
2.3.1.2	Eviter les discontinuités sans ajouter d'effet de flou	130
2.3.1.3	Eviter la perte de couleur	134
2.3.2	Génération d'une texture infinie cylindrique	136
2.3.3	Exemples d'écorces synthétisées	137
2.4	Conclusion	138
Chapitre 3	Application de texture aux branches	140
3.1	Introduction	140
3.2	Surfaces implicites	143
3.2.1	Choix du volume de base	143
3.2.2	Adoucissement des jonctions par mélange de potentiels (« <i>blending</i> »)	146
3.3	Application de la texture d'écorce	152
3.3.1	Paramétrisation des coordonnées de texture	152
3.3.1.1	Calcul de la coordonnée u	155
3.3.1.2	Calcul de la coordonnée v	156
3.3.1.3	Remarques	159
3.3.2	Déplacement de surface (« <i>displacement mapping</i> »)	160
3.4	Application par distribution d'éléments géométriques	161

3.5	Conclusion.....	163
Chapitre 4	Les feuilles.....	166
4.1	Introduction	166
4.2	Représentation géométrique des feuilles.....	166
4.3	Disposition des feuilles sur l'arbre.....	169
4.3.1	Instanciation, orientation et positionnement	169
4.3.2	Gestion des intersections de feuilles	171
4.4	Rendu des feuilles	172
4.4.1	Affichage 3D accéléré matériellement.....	172
4.4.2	Lancer de rayons	172
4.5	Détection des nervures à partir d'une photo.....	173
4.5.1	Motivation	173
4.5.2	Principe de la méthode	175
4.6	Translucidité et réflectance	178
4.7	Conclusion.....	179
Chapitre 5	Résultats	180
	Conclusion.....	184
	Bibliographie.....	188

Liste des figures

- Figure 1** : A gauche : une image binaire d'une théière en fil de fer illustrant les débuts de l'infographie. A droite : une très belle image « photo-réaliste », synthétisée par l'artiste Chris Pember, donnant un exemple de ce que l'on peut produire de nos jours en Infographie..... 22
- Figure 2** : Exemples des différents domaines d'application des arbres en images de synthèse. De haut en bas et de gauche à droite : urbanisme (« Area Oviedo », ©2001 Luis Rivero / Onyx Computing) ; environnement (« Tremblay-Vieux-Pays », ©2003 CIRAD) ; cinéma (« A bug's life », ©1998 Disney) ; jeu vidéo (« Ghost Recon », ©2001 Ubi Soft). 25
- Figure 3** : Vues rapprochées d'arbre et de feuille dans le film d'animation « A bug's life ». Des artistes ont modélisé la scène et ont dessiné et appliqué les textures manuellement..... 25
- Figure 4** : Vue d'ensemble de la méthode visant à appliquer une texture d'écorce en relief sur des branches d'arbre à partir d'une photo d'écorce, d'un modèle d'arbre et de quelques réglages simples, suivant les méthodes d'analyse employées..... 28
- Figure 5** : Schéma d'un arbre feuillu quelconque. On retrouve toutefois les mêmes composantes dans l'autre catégorie d'arbres que sont les conifères (cf. paragraphe 1.2.1.2). 33
- Figure 6** : Bourgeons et feuilles sur un rameau. 36
- Figure 7** : Types de croissance des nouvelles branches. A gauche : croissance orthotropique. A droite : croissance plagiotropique..... 37
- Figure 8** : Vue en coupe simplifiée d'un tronc ou d'une branche..... 38
- Figure 9** : Classification des écorces suivant des critères visuels..... 40
- Figure 10** : Les différentes nervations. De gauche à droite : pennée, parallèle, palmée. 42

Figure 11 : Différentes dispositions des feuilles sur une branche. De gauche à droite : alternes, opposées, décussées, verticillées.....	42
Figure 12 : Quelques exemples, très loin d'être exhaustifs, illustrant la grande diversité des de feuilles. Sur fond blanc : feuilles d'arbres « feuillus ». Sur fond noir : épines de « conifères ».....	43
Figure 13 : Panneaux en croix pour donner une impression de volume et permettre de se déplacer autour de l'arbre.....	46
Figure 14 : Exemples d'arbres extraits du monde des jeux vidéo. En haut : « Medal of Honor : Allied Assault »(jeu, ©2002 Electronic Arts). En bas : « CodeCreatures Benchmark » (logiciel principalement destiné aux joueurs pour tester les performances de leur PC, ©2002 CodeCult). Les branches principales ont du volume alors que les feuilles et les petites branches sont remplacées par des textures 2D transparentes.....	48
Figure 15 : Image extraite du film « The adventures of André and Wally B. » réalisé par W. T. Reeves et R. Blau.....	49
Figure 16 : squelette d'un arbre représenté en « fils de fer ».....	50
Figure 17 : Exemple simple de L-système dont l'axiome est A. On peut assimiler A à un bourgeon et S à un segment d'entre-nœud.....	52
Figure 18 : Exemple de représentation d'arbre avec une matrice de ramification.....	54
Figure 19 : Deux exemples d'arbres obtenus par Deussen et Lintermann.....	56
Figure 20 : Schéma d'un arbre représenté avec des cylindres.....	58
Figure 21 : Arbres formés de troncs de cône. A gauche : chaque tronc de cône a des bases parallèles, ce qui engendre des « trous ». A droite : les troncs de cônes consécutifs ont des bases communes afin de supprimer les discontinuités d'un segment de branche à l'autre.....	59
Figure 22 : A gauche : un cône-sphère. A droite : branche formée de cône-sphères.....	60
Figure 23 : Cylindres généralisés. Celui de droite illustre un cas particulier où le potentiel d'un point P ne correspond pas au point de la courbe le plus proche.....	61

Figure 24 : Cylindre généralisé discrétisé.....	62
Figure 25 : Photographies d’embranchements sur de vrais arbres. A gauche : une photo d’embranchement et une image illustrant sa silhouette où l’on voit clairement la transition adoucie entre les branches. A droite : un autre exemple d’embranchement où les larges arcs blancs montrent les zones nécessitant d’être adoucies par rapport aux lignes oranges qui montrent les contours qu’aurait l’embranchement sans adoucissement.	63
Figure 26 : Principe du « ray marching » sur une surface implicite quelconque.	65
Figure 27 : Juxtaposition d’une texture 2D (appliquée quatre fois) qui ne « boucle » pas (à gauche), puis qui « boucle » (à droite). Dans le premier cas, on constate des discontinuités.	66
Figure 28 : branches recouvertes d’une texture 2D d’écorce dont l’aspect lisse est peu réaliste.	67
Figure 29 : Le « Puissant Erable » (« Mighty Maple ») de Jules Bloomenthal [12] dont la texture d’écorce est obtenue par perturbation de normale.	68
Figure 30 : Jonction par mélange de deux branches implicites (Hart [50]).....	69
Figure 31 : Résultats obtenus par Hart et Baker en recouvrant des branches avec des texels.....	70
Figure 32 : Divers résultats obtenus avec la méthode de synthèse de déchirures de Lefebvre et Neyret.....	71
Figure 33 : Feuilles modélisées à partir de textures détournées plaquées sur un ensemble réduit de polygones.	74
Figure 34 : Simulation de la transmission de la lumière à travers des feuilles.....	74
Figure 35 : A gauche : arbre virtuel observé de loin pour lequel les détails de l’écorce ne sont pas importants. Au centre : vue rapprochée (et, en médaillon, encore plus rapprochée) sur des branches d’un arbre texturé avec un placage de textures 2D traditionnel. A droite : un résultat significativement amélioré grâce à notre technique utilisant de l’écorce en relief (et une transition adoucie aux intersections de branches que nous présentons au Chapitre 3).	77

- Figure 36** : Illumination d'une carte de hauteurs (connue) représentant une texture d'écorce (une par ligne) afin de comparer les ressemblances entre la carte de hauteur et le résultat illuminé. (a) Modèle connu (photographie d'écorce) utilisé comme carte de hauteurs. (b) Illumination directionnelle avec une source lumineuse venant du coin haut-gauche. (c) Même illumination que (b) mais vue de face. (d) Même vue que (c) mais avec une élévation plus basse de la texture pour simuler un élargissement (relatif) des craquelures. (e) Illumination globale diffuse avec la même élévation que sur les images (b) et (c). (f) Même vue que (e) mais avec la même élévation que (d). 83
- Figure 37** : Trois paires de photos de trois écorces différentes. Chaque paire est composée d'une photographie prise dans de mauvaises conditions (à gauche) et d'une autre correcte (à droite). Les problèmes illustrés ici sont : (a) éclairage partiel ; (b) saleté ; (c) lichen..... 88
- Figure 38** : Deux exemples d'utilisation directe du modèle d'écorce en niveaux de gris comme carte de hauteurs. Les deux reliefs de l'exemple de droite montrent d'abord la carte de hauteurs utilisée telle quelle puis la carte inversée pour convenir à cette écorce de bouleau. Note : les vues monochromes en 3D avec déplacement de surface ont été éclairées avec une lumière venant de la direction en haut à gauche. 90
- Figure 39** : Fonctions de saturation (valeurs entre 0 et 1 sur les abscisses et les ordonnées). De gauche à droite : niveau haut ; niveau bas ; niveaux haut et bas ; trois niveaux..... 93
- Figure 40** : Séparation des deux principaux niveaux de hauteur de la texture d'écorce (a) par augmentation du contraste, d'abord sans prendre en compte les tâches sombres mais néanmoins à la surface de l'écorce (b), puis en en tenant compte (c). 93
- Figure 41** : Les deux exemples de la Figure 38, avant et après réduction du bruit. La texture de gauche a été traitée par filtrage médian tandis que la texture de droite a été traitée par saturation du niveau bas, d'où l'impression de contraste plus fort qui est simplement dû à l'étirement des profondeurs entre les extrema du relief. 94

- Figure 42** : Quelques écorces illustrant les différentes catégories de notre classification. Chaque écorce est présentée par couple (modèle et, en dessous, carte de hauteur que nous avons extraite). Chaque cadre englobe une catégorie comportant un ou plusieurs exemples. De gauche à droite et de haut en bas, on trouve les catégories de 1 à 6. Le dernier cadre en bas à droite contient d'autres écorces difficilement traitables par notre méthode. 98
- Figure 43** : Segmentation du modèle (a) en plusieurs couleurs pour faire ressortir les couleurs principales que l'on regroupe en deux niveaux (g). Les cinq images du milieu (b–f) représentent, de gauche à droite, une quantification en 2, 3, 4, 5 et 6 couleurs. Dans cet exemple, ce n'est qu'à partir de 5 couleurs qu'on commence à voir ressortir le fond des craquelures de couleur marron et il faut donc quantifier l'image en 6 couleurs pour ne pas retrouver trop de pixels marrons dans le reste de la texture. L'image segmentée (g) a été formée en regroupant manuellement les couleurs de l'image (f) en deux catégories..... 102
- Figure 44** : Segmentation à partir d'une quantification sur la teinte uniquement. De gauche à droite : le modèle ; suppression de la luminosité ; quantification en deux couleurs ; image segmentée en affectant un niveau à chaque couleur. 102
- Figure 45** : De gauche à droite : 1°) le modèle ; 2°) segmentation par quantification en deux couleurs ; 3°) quantification en trois couleurs ; 4°) regroupement des deux couleurs les plus claires de la quantification en trois couleurs..... 103
- Figure 46** : La texture d'écorce (à gauche) et a été segmentée en deux niveaux (au milieu) puis a subi une fermeture, une dilation, une transformée de distance et enfin un lissage (léger flou) pour obtenir la macro-texture (à droite). 105
- Figure 47** : Estimation de la proportion a / A de la micro-texture (d'amplitude a) par rapport à la macro-texture (d'amplitude A). Dans cet exemple la proportion peut être estimée à $1/3$. La ligne en trait gras représente une vue en coupe d'une bosse de la macro-texture. La ligne continue plus fine et chaotique représente le véritable relief de la texture que l'utilisateur doit estimer visuellement. On note que l'amplitude du relief réel est donc $A+a$ 107

Figure 48 : Etude des spectres de puissance d'une carte de hauteurs et de son relief éclairé (la lumière provient d'en haut à gauche) afin de déterminer quel filtre permettrait de retrouver le relief initial à partir de l'image éclairée. Chaque ligne présente une texture différente. Chaque paire d'images représente une texture et son spectre de puissance obtenu par transformée de Fourier discrète.....	108
Figure 49 : Filtre Gaussien pour le domaine spectral.	109
Figure 50 : A gauche : filtre anisotropique perpendiculaire à la direction \vec{L} de la source lumineuse. A droite : α_L est l'angle entre \vec{L} et la normale \vec{N} au plan de base de la texture.	110
Figure 51 : Série de filtres pour des angles α_L décroissant de $\pi/2-\varepsilon$ (lumière rasante) à $0+\varepsilon$ (lumière proche du zénith).	111
Figure 52 : Deux exemples d'arbres dont le relief de la texture d'écorce a été extrait grâce à notre méthode (le modèle et le relief sont en haut à droite de chaque exemple).	113
Figure 53 : Deux exemples de segmentation en zones représentatives de la macro-texture. La texture d'écorce du deuxième exemple nécessite une segmentation en trois classes.....	115
Figure 54 : Filtres linéaires pour une image sans orientation de texture (à gauche) et avec orientation horizontale (à droite).....	116
Figure 55 : a) modèle ; b) segmentation correspondante en deux niveaux ; c) spectre de puissance (obtenu par transformée de Fourier discrète) donnant les dimensions du masque (le ratio calculé ici vaut environ 3, ce qui signifie que la texture est très orientée horizontalement) ; d) filtrage de l'image segmentée, avec le masque de convolution calculé.	118
Figure 56 : Courbes de profil dépendant du type d'écorce. A gauche : identité (on ne modifie pas le profil). Au centre : sigmoïde (permet d'obtenir un profil doux). A droite : transition abrupte (permet de représenter des transitions abruptes tout en ayant réduit les pixels parasites grâce au filtrage par convolution).....	118

- Figure 57** : Création d'un échantillon de base E_0 qui boucle. Les pointillés sur les images (a) à (c) représentent les dimensions des échantillons et les rectangles continus représentent la partie opaque des morceaux de texture dupliqués. a) Collage de morceaux de même taille sans mélange (les morceaux sont numérotés dans l'ordre où ils sont collés) ; b) ajout d'une valeur aléatoire à la taille ; c) rajout des zones de mélange (solution retenue) ; d) exemple de modèle ; e) l'échantillon de base E_0 correspondant au modèle et qui « boucle »..... 129
- Figure 58** : Duplication des morceaux de texture (représentés par des lettres dans l'ordre où ils sont collés sur l'image de gauche) au centre de chaque échantillon E_i initialisés avec E_0 pour synthétiser différents échantillons d'une même texture. Les quatre échantillons synthétisés (à droite) sont visuellement différents mais leurs bordures sont identiques..... 129
- Figure 59** : Opacité de masque de mélange selon $dist_{opaque}$ dans la ZTM. A gauche : la zone grisée représente la ZEM. A droite : quatre exemples de positionnement de la ZEM dans la ZTM pour différentes valeurs de $dist_{couleur}$ (les courbes les plus à gauche représentent le mélange de pixels contrastés)..... 134
- Figure 60** : Deux exemples de masques de mélange quelconques qui ont servi lors de la synthèse d'échantillons à partir du modèle d'écorce. Note: Dans cette figure, les masques ont été grossis par rapport aux modèles mais leur largeur est normalement légèrement supérieure à un quart de la largeur du modèle. Chaque pixel du masque dépend du pixel du morceau de texture à coller et du pixel de l'échantillon avec lequel il sera mélangé..... 134
- Figure 61** : Différentes étapes du processus de remplissage Markovien (linéaire) de la table contenant les indices des échantillons de textures synthétisés. La case en cours de traitement (en gras) doit recevoir un indice différent de ceux déjà assignés (cases grisées) à ses voisins (les liens indiquent les voisins à prendre en compte)...... 136
- Figure 62** : Quatre écorces différentes synthétisées à partir de leurs modèles (cartes de couleurs et de hauteurs)..... 137

Figure 63 : A gauche : gonflement couramment rencontré à la jonction de deux objets implicites mélangés et de dimensions proches. A droite : si l'un des deux objets cylindriques a un champ de potentiel de rayon inférieur à celui de l'autre objet, alors il n'y a pas de gonflement.	141
Figure 64 : A gauche : un volume implicite cylindrique. A droite : un cône-sphère.	144
Figure 65 : Schéma d'un cône-sphère (contour en gras) en deux dimensions.	144
Figure 66 : Embranchements de surfaces implicites mélangées. A gauche : embranchement simple. A droite : embranchement à plus de deux branches filles.	148
Figure 67 : Contrainte de longueur minimale des segments par rapport au rayon du champ de potentiels. A gauche la longueur des segments est très largement suffisante. Au centre : la longueur devient critique mais les zones de mélange n'étant pas du même côté, il n'y a pas de problème. A droite : il y a un problème d'intersection des zones de mélange quand elle est située du même côté d'un segment trop court.	148
Figure 68 : Importance de la non-linéarité des potentiels. A gauche : les potentiels sont linéaires. Au centre : ils sont transformés par la fonction h (à droite).....	149
Figure 69 : Paramétrisation de la texture sur des branches. La texture suit les branches de même diamètre mais elle est mélangée à la jonction de branches de tailles différentes. Sur l'image de droite, on remarque la discontinuité de texture à l'intérieur de l'embranchement. Elle devient toutefois beaucoup moins visible avec une texture d'écorce rugueuse.....	153
Figure 70 : Isolement de chaque segment de branche par des plans médians (en pointillés). Les petites branches (ici en vert) sont simplement mélangées. Les zones en gris clair et gris foncé montrent la continuité de la texture le long des branches à l'exception de l'intérieur d'un embranchement de branches de même diamètre.	154

Figure 71 : Lors de la modélisation du squelette de l'arbre, la rotation autour de l'axe z de deux segments (cône-sphères) consécutifs n'est pas autorisée si elle est différente de $2\pi/N_{ETB}$ (dans cet exemple, $N_{ETB} = 4$). A gauche : cas interdit. Au centre et à droite : cas autorisés. Les flèches montrent l'axe x utilisé comme base pour l'angle α donnant la position de P autour de l'axe z.....	156
Figure 72 : Les sommets représentant des bases de branches (à la naissance des branches de diamètre inférieur à la branche parente) sont représentés ici par des points circulaires. Les branches de niveau 1 sont en noir et leur base est le point noir (unique) à la base du tronc. Les branches de niveau 2 sont en gris foncé et celles de niveau 3 en gris clair.....	157
Figure 73 : Déformation de la texture grâce à des plans médians séparant le segment en trait gras de ses voisins.	157
Figure 74 : Répartition d'épines sur un tronc.	162
Figure 75 : Etapes d'acquisition de chacune des deux textures des faces d'une feuille. Les pointillés de l'étape 6 délimitent la zone sélectionnée.	168
Figure 76 : Les différents repères utilisés pour les branches et les feuilles de l'arbre. De gauche à droite : arbre, segment de branche terminal (pétiole ou rameau), feuille.....	170
Figure 77 : Détection de la face (la normale $\vec{N} = \vec{AB} \wedge \vec{AC}$ est sur la face supérieure) visée par un rayon \vec{R} . Angle $> 90^\circ \Rightarrow$ face supérieure. Angle $< 90^\circ \Rightarrow$ face inférieure.	173
Figure 78 : Comparaison entre des feuilles texturées par simple placage 2D (à gauche) et avec perturbation de normales à partir d'une carte de relief (à droite). Les zones cerclées font ressortir les principales améliorations apportées par cette seconde méthode.	174
Figure 79 : Deux formes de section de nervure de feuille. Le premier cas est plus courant.....	174
Figure 80 : Extraction du relief des deux faces d'une feuille à partir de la face inférieure.	175

Figure 81 : Deux autres exemples de feuilles dont nous avons extrait le relief des deux faces à partir de la face inférieure.	177
Figure 82 : Photographies en gros plan de nervures de feuilles vues à contre jour.....	178
Figure 83 : Un érable virtuel.....	180
Figure 84 : Deux exemples d'arbres virtuels.....	181
Figure 85 : Deux autres arbres virtuels avec des textures d'écorce peu adaptées aux méthodes que nous avons proposées. L'écorce de l'image du haut ne se prête pas au mieux à notre synthèse de texture qui introduit un effet de flou, bien qu'elle soit censée le limiter. Dans l'image du bas, la texture contient des écailles qui se chevauchent et dont la luminosité varie indépendamment de l'éclairage, ce qui est un cas que notre méthode ne gère pas très bien. Les images générées offrent néanmoins un résultat intéressant.....	182

Introduction

L'Infographie

Depuis les débuts de l'Infographie, les chercheurs ont tenté de représenter la complexité du monde qui nous entoure. Cela va des êtres vivants aux paysages, incluant le ciel, les montagnes et les végétaux. En effet, que ce soit pour mieux comprendre notre environnement, pour le simuler ou bien dans un but purement artistique, il est important de pouvoir reproduire, le plus fidèlement possible, le monde réel sur ordinateur.

Dans le domaine artistique, au sens large du terme, qui est un des domaines les plus demandeurs en images de synthèse « photo-réalistes », on pensera en premier lieu au monde du cinéma qui a besoin d'intégrer des images de synthèse dans des séquences filmées afin de créer par exemple des effets impossibles, trop coûteux ou trop dangereux à filmer en vrai (des arbres qui marchent, des immeubles qui explosent, des tornades qui ravagent une ville...). Vient ensuite le milieu du jeu vidéo qui fait aujourd'hui de plus en plus appel aux mêmes outils que ceux employés pour les effets spéciaux du Septième Art.

Mais que signifie exactement « photo-réalisme » ? Ce terme est le plus couramment utilisé dans le sens où un œil non-averti – ou même, dans l'idéal, averti – ne saurait dire si l'image de synthèse qu'il regarde en est une ou bien si c'est une photographie. Cependant, ce critère ne caractérise que l'aspect visuel de l'image. Une scène peut parfaitement avoir l'air réaliste sans pour autant l'être physiquement. L'observateur le remarquera si des règles qu'il connaît ont été enfreintes ; mais, dans le cas contraire il jugera la scène photo-réaliste. L'exemple suivant montre que la limite à partir de laquelle une image peut être jugée photo-réaliste est très subjective. Supposons que l'on soit en mesure de représenter en images de synthèse tous les éléments qui composent un arbre. Si l'on présente à quelqu'un qui n'est pas très érudit en botanique, une image de chêne auquel on aura donné des feuilles d'érable, alors cette personne dira probablement que ça ressemble beaucoup à une photo d'arbre, voire à une photo d'érable s'il a su reconnaître les feuilles. La même image présentée à un spécialiste sera sans doute jugée truquée. Néanmoins, si ce spécialiste

a du mal à dire si le trucage a été fait en images de synthèse ou à partir d'un montage de photographies de chêne et d'érable, alors on pourra dire malgré tout que l'image est photo-réaliste.

La Figure 1 illustre les progrès saisissants de l'Infographie qui nous permettent aujourd'hui d'obtenir des résultats très impressionnants en termes de réalisme visuel. Ce genre de résultat est cependant certainement inférieur à ce que l'on pourra atteindre dans les prochaines années. En effet, bien que la qualité de restitution des images calculées sur ordinateur soit aujourd'hui potentiellement excellente, les ressources (en puissance de calcul et en espace mémoire) nécessaires à l'obtention d'images que l'on puisse qualifier, dans l'absolu, de « photo-réalistes » sont encore trop souvent insuffisantes, bien que nettement supérieures à ce que l'on pouvait faire, il y a quelques années.



Figure 1 : A gauche : une image binaire d'une théière en fil de fer illustrant les débuts de l'infographie. A droite : une très belle image « photo-réaliste », synthétisée par l'artiste Chris Pember, donnant un exemple de ce que l'on peut produire de nos jours en Infographie.

D'autre part, l'obtention de tels résultats nécessite le plus souvent un lourd travail artistique. Il faut donc assister au maximum les artistes dans leur tâche, ce qui fait resurgir les problèmes au niveau de la puissance des outils informatiques que l'on peut leur proposer. Ces contraintes matérielles sont encore plus importantes dans le domaine de la simulation de phénomènes naturels (croissance des plantes, mécanique des fluides, etc.) qui sont souvent parmi les plus complexes car, contrairement à ce qui a été conçu par l'Homme et qui est donc conditionné à son mode de réflexion, la Nature ne connaît pas de telles limites et ne peut être comprise qu'en la simplifiant, en isolant chaque phénomène, alors qu'ils sont tous plus ou moins liés. Enfin, quand bien même disposerait-on des

ressources nécessaires à la simulation de tels phénomènes, il faudrait encore être en mesure de découvrir les modèles physiques, puis numériques, qui les régissent et enfin les algorithmes pour les résoudre. Ainsi, les chercheurs ont tenté de comprendre et de simplifier ces phénomènes pour que leur simulation informatique soit possible.

Le cadre de la thèse

Parmi les nombreux éléments naturels nous environnant, l'un des plus importants est sans aucun doute la végétation. Et ce sont tout particulièrement les arbres et les arbustes auxquels nous nous intéressons ici, même si un certain nombre des notions traitées peuvent être également employées pour les plantes. De plus, nous avons principalement orienté nos recherches sur les détails visibles lorsqu'on observe les arbres de près. Nous avons déjà mentionné l'utilité des images de synthèse pour représenter informatiquement notre environnement. Il est évident que ce principe se rapporte également aux arbres qui en font partie intégrante, jusqu'au cœur des zones urbaines. Les principaux champs d'application, illustrés dans la Figure 2, sont :

- **L'architecture et l'urbanisme.** « Dans une ville, un quartier sans arbres paraît sévère et sans charme tandis que là où les arbres prospèrent, ils contribuent à une atmosphère sereine » [107]. Les architectes et les urbanistes en étant parfaitement conscients, ils ne manquent donc pas d'intégrer des arbres et des arbustes à leurs projets. Etant donné que ces derniers sont de plus en plus souvent réalisés sur ordinateur, afin d'offrir aux clients des visites virtuelles des ouvrages, permettant ainsi éventuellement de procéder à des modifications avant la construction définitive, il est nécessaire de pouvoir modéliser sur ordinateur les espèces d'arbres que l'on voudra y faire pousser. Cependant, ce domaine n'est pas encore aujourd'hui le plus demandeur en photo-réalisme car les clients ont très longtemps été habitués à ne voir que des croquis et dessins sur papier et la demande en réalisme se porte toujours plus sur les bâtiments et leur éclairage que sur la végétation environnante.
- **L'environnement.** Un aspect à ne pas négliger dans la visualisation des arbres est la simulation de leur croissance. Si cela est important pour obtenir des arbres visuellement réalistes, c'est capital dans les domaines tels que la gestion de l'environnement, la sylviculture, l'aménagement du territoire, la recherche agronomique, etc. Cependant,

ces domaines ne sont pas encore très demandeurs en termes de photo-réalisme et un rendu correct des arbres vus d'une certaine distance est souvent suffisant.

- **Le cinéma.** Au cinéma, les arbres en images de synthèse sont utilisés dans des films comme, par exemple, *Twister* (©1996 : Amblin / Universal Pictures / Warner Bros / Constant C) où l'on peut voir des arbres virtuels déracinés par une tornade, elle aussi virtuelle. Un autre exemple récent fut l'intégration dans le film *The Lord of the Rings : The two towers* (©2002 : New Line Cinema / Wingnut Films) des « Ents », êtres vivants imaginaires à la forme d'arbres issus de l'univers fantastique de J.R.R. Tolkien. Mais les films classiques ne sont pas les plus demandeurs en arbres virtuels. Ces derniers sont surtout utilisés dans les films d'animation en images de synthèse qui ont besoin de recréer des personnages et tout leur environnement ; et qui doivent donc, presque inévitablement, faire appel à la synthèse d'arbres. Les exemples ne manquent pas et pour n'en citer que quelques-uns, on trouve : *A Bug's Life* (Pixar / Disney, 1998), *Antz* (©1998 : Dreamworks / Pacific Data Images), *Shrek* (©2001 : Dreamworks / Pacific Data Images), *Toy Story* (©1995 : Pixar / Disney) et *Toy Story 2* (©1999 : Pixar / Disney). Même les dessins animés classiques font aujourd'hui appel à l'infographie pour représenter des arbres. C'est notamment le cas de *Tarzan* (©1999 : Disney) qui insère habilement des arbres en 3D dans des scènes 2D.
- **Les jeux vidéo.** On distingue deux aspects différents dans les jeux vidéo : d'une part les séquences cinématiques pré-calculées et autres décors fixes qui peuvent être rendus à partir des mêmes outils que ceux employés dans l'industrie du cinéma ; et, d'autre part, le rendu 3D en temps réel dans le jeu lui-même. En effet, la plupart des jeux actuels permettent de déplacer un ou plusieurs personnages, vus à la première ou à la troisième personne, dans des décors divers et variés où l'on rencontre presque inévitablement des arbres. Dans ce cas, le plus gros effort est porté sur le compromis entre le souci du détail, qui croît au rythme de l'avancée technologique des nouvelles cartes graphiques, et la légèreté (en termes de polygones à afficher) requise pour que le jeu s'exécute de la façon la plus fluide possible, sans consommer les ressources nécessaires pour le reste du jeu (affichage du reste des décors, intelligence artificielle des personnages, simulation des modèles physiques, etc.). Cet aspect des jeux vidéo se retrouve également dans les simulateurs de vol et de conduite, que l'on peut assimiler aux jeux vidéo tant les similitudes sont grandes sur les techniques de visualisation.



Figure 2 : Exemples des différents domaines d'application des arbres en images de synthèse. De haut en bas et de gauche à droite : urbanisme (« Area Oviedo », ©2001 Luis Rivero / Onyx Computing) ; environnement (« Tremblay-Vieux-Pays », ©2003 CIRAD) ; cinéma (« A bug's life », ©1998 Disney) ; jeu vidéo (« Ghost Recon », ©2001 Ubi Soft).



Figure 3 : Vues rapprochées d'arbre et de feuille dans le film d'animation « A bug's life ». Des artistes ont modélisé la scène et ont dessiné et appliqué les textures manuellement.

Toutefois, toutes les productions de ces deux derniers domaines font grandement appel au talent des artistes qui doivent parfois modéliser la structure des arbres manuellement afin d'obtenir la forme désirée ; mais qui, surtout, concentrent leurs efforts sur les détails des arbres lorsque ceux-ci sont vus de près (voir Figure 3 montrant diverses scènes du film d'animation « *A bug's life* », ainsi que les deux images sur la droite de la Figure 14 au paragraphe 1.3.1.2). Il est donc intéressant de leur proposer des outils leur permettant de générer plus facilement des arbres ; idéalement : sans qu'ils aient le moindre effort à fournir afin de consacrer leurs talents à des fins plus créatives que la simple modélisation d'arbres.

De nombreux travaux ont été effectués dans le domaine de la génération d'arbres, en s'intéressant plus particulièrement au développement des branches. Toutefois, la plupart de ces travaux donnent de très bons résultats visuels à une certaine distance mais souffrent d'un manque de réalisme lorsque l'observateur se rapproche. Pour synthétiser un arbre de façon réaliste sur ordinateur, il est nécessaire de bien connaître et comprendre sa structure et la façon dont celle-ci se développe. Mais, la structure seule ne suffit pas ; il faut également que tout ce que l'on voit extérieurement de l'arbre (l'écorce, les feuilles, etc.) soit – ou ait l'air – réaliste.

D'autre part, une fois que l'arbre est correctement représenté en données numériques, il risque de nécessiter une place très importante en mémoire, surtout si l'on souhaite modéliser une forêt. Il faut donc également, dans la mesure du possible, limiter la consommation de mémoire.

Le sujet de cette thèse consiste donc à modéliser et visualiser en images de synthèse des arbres de la manière la plus visuellement réaliste possible, et ceci même lorsqu'on les observe de près, tout en essayant de limiter l'utilisation des ressources matérielles (puissance de calcul et mémoire) lorsque cela est possible sans que la qualité des résultats en pâtisse.

Les apports de nos travaux

Les apports les plus importants de nos travaux se portent sur l'écorce des arbres, grâce à des méthodes permettant de recouvrir un arbre avec une texture réaliste d'écorce en

relief. En effet, lorsqu'on observe un arbre de près, on remarque la rugosité de l'écorce sur le tronc et les branches. Constatant que l'utilisation de photographies permet de maximiser le réalisme visuel, et dans un souci d'offrir à l'utilisateur la solution la plus pratique possible, nous proposons un moyen de synthétiser une texture comportant couleur et relief, à partir d'une simple photographie d'écorce, pour une grande majorité des espèces d'arbres. L'analyse du relief d'une texture à partir d'une photographie étant difficile, nous proposons plusieurs méthodes d'extraction de ce relief afin de gérer un maximum de cas. D'autre part, pour qu'on ne puisse pas constater de discontinuités lors de l'application de cette texture sur les branches, nous avons élaboré une méthode de synthèse de texture infinie à partir de l'échantillon de texture en relief, tout en évitant l'effet de répétition de motif. De cette façon, nous obtenons des résultats très satisfaisants en termes de réalisme visuel pour une grande majorité de types d'écorces, sans nécessiter d'effort particulier de la part de l'utilisateur qui n'aura qu'à fournir un modèle d'arbre, une photographie d'écorce et quelques paramètres simples comme la catégorie à laquelle appartient l'espèce d'arbre parmi une liste que nous avons établie, permettant de définir la méthode d'analyse appropriée et ses paramètres.

Nous avons également développé une méthode de modélisation du volume des branches afin d'y faciliter l'application de textures d'écorce. Les jonctions de ces branches sont lissées tout en conservant l'aspect rugueux de l'écorce, afin de ressembler au mieux à la réalité. Les volumes permettant d'effectuer des jonctions douces entre deux objets sont les surfaces implicites. Cependant, elles sont généralement utilisées pour des objets lisses et posent parfois des problèmes de gonflements lorsqu'on mélange deux objets de mêmes dimensions tels que des segments de branches. Nous présentons donc un moyen de mélanger, avec une transition douce, des objets (segments de branches) rugueux comme l'écorce sans gonflement à leurs jonctions. En outre, la texture infinie en relief que nous synthétisons étant basée sur un cylindre (le relief étant une carte de hauteur en chaque point du cylindre), et la surface de l'ensemble des branches d'un arbre étant plus complexe qu'un simple cylindre, nous avons donc proposé une méthode simple et peu coûteuse en mémoire pour appliquer cette texture sur des branches d'arbre. Nous présentons aussi une autre méthode additionnelle pour traiter les quelques écorces comportant des formes en relief, telles que des épines, qui ne peuvent pas être représentées par une carte de relief.

La Figure 4 présente une vue d'ensemble de la méthode d'analyse et de synthèse d'écorce en relief en vue de l'appliquer sur des branches d'arbres définies par une surface implicite.

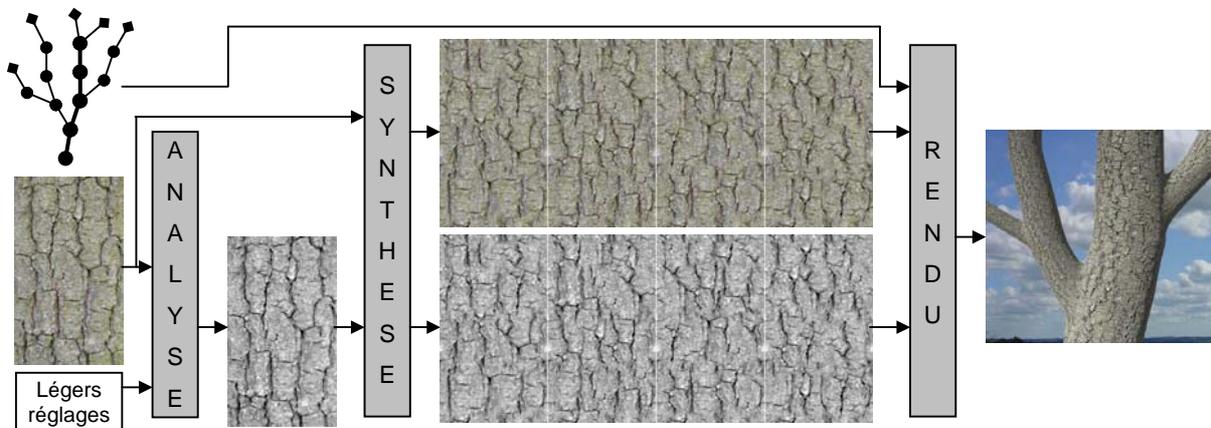


Figure 4 : Vue d'ensemble de la méthode visant à appliquer une texture d'écorce en relief sur des branches d'arbre à partir d'une photo d'écorce, d'un modèle d'arbre et de quelques réglages simples, suivant les méthodes d'analyse employées.

Enfin, une méthode consistant à analyser une photographie de feuille numérisée a été proposée pour extraire les informations de relief des deux faces de cette feuille. Le relief, généré par des traitements d'images simples à partir de la photographie fournie, est alors utilisé lors du rendu pour l'améliorer en reproduisant le relief des nervures des feuilles et peut aussi servir à altérer leur translucidité en distinguant les nervures du reste du limbe.

Organisation du mémoire de la thèse

Dans ce manuscrit, nous commençons par présenter un état de l'art de la représentation d'arbres sur ordinateur. Ce chapitre débute par une vue d'ensemble des notions de Botanique qui peuvent servir à une meilleure compréhension du sujet traité. S'ensuivent les différentes méthodes qui ont vu le jour pour représenter des arbres en images de synthèse, en mettant l'accent sur la modélisation des branches, l'application de textures d'écorce et le rajout de feuilles. Ces trois domaines – et surtout ceux qui traitent de l'écorce, nécessitant encore un certain nombre d'améliorations pour atteindre le photo-

réalisme souhaité ou, au moins, s'en rapprocher – bénéficient donc chacun d'un des trois chapitres suivants.

Le Chapitre 2 traite du sujet principal de cette thèse : l'écorce. Il est, d'une part, consacré à l'extraction du relief à partir d'une photographie quelconque d'écorce et, d'autre part, à la synthèse d'une texture infinie d'écorce en relief à partir de cette photographie et des informations de relief qui en auront été extraites.

Le Chapitre 3 explique comment modéliser et joindre des segments de branche avec des surfaces implicites, ainsi que comment y appliquer les textures d'écorces synthétisées au chapitre précédent.

Le Chapitre 4 expose les quelques améliorations que nous apportons à la modélisation des feuilles et à l'extraction de leurs nervures, toutes deux à partir d'une photographie afin de nous assurer un maximum de chances d'obtenir un rendu photo-réaliste.

Enfin nous présentons quelques images d'arbres que nous avons synthétisées afin d'illustrer l'ensemble des améliorations que nous avons apportées. Puis nous donnons nos conclusions sur le travail effectué et sur ce qui reste encore à faire dans le domaine du rendu d'arbres, dans le but final d'obtenir des images photo-réalistes, même de près.

Chapitre 1

Etat de l'art sur la représentation d'arbres en images de synthèse

1.1 Introduction

Une règle évidente, et néanmoins fondamentale, lorsqu'on veut simuler un élément ou un phénomène naturel, est qu'il faut au préalable l'étudier, chercher à comprendre comment il est composé, comment il évolue, quelles sont les règles qui le régissent, etc. Alors, seulement une fois ce travail effectué, on peut tenter de représenter le sujet étudié par un modèle numérique. C'est pour cette raison qu'avant d'étudier les différentes méthodes existant à ce jour pour représenter des arbres en images de synthèse, nous consacrons d'abord le paragraphe 1.2 suivant aux notions de Botanique nécessaires pour bien comprendre le sujet que nous allons traiter.

Nous procédons ensuite à l'inventaire des différentes méthodes qui ont été utilisées pour représenter des arbres synthétiques en Informatique graphique, depuis les débuts de ce thème jusqu'à aujourd'hui. Le sujet de cette thèse étant plus précisément la représentation virtuelle d'arbres vus de près, c'est-à-dire avec un certain souci du détail, nous ne nous attachons pas simplement aux méthodes qui ont été employées pour modéliser la structure des arbres, mais surtout aux méthodes passées et actuelles pour représenter les détails des arbres : principalement l'écorce et dans une moindre mesure les feuilles. Nous ne traitons ni des fleurs ni des fruits dans cette thèse car ce sont des éléments secondaires qui n'interviennent qu'à certaines périodes de l'année et qui ne sont donc pas systématiquement présents sur les arbres, contrairement à l'écorce (toujours) ou encore les feuilles (la plupart du temps). Notons toutefois que certains arbres comme les sapins n'ont pas à proprement parler de feuilles mais de très nombreuses aiguilles ou écailles si denses qu'elles cachent la majeure partie de l'écorce des branches.

Après le recensement de ce qui a déjà été fait dans ce domaine, nous pourrons alors dresser une liste des difficultés rencontrées avec les méthodes actuelles et de ce qui peut être fait pour améliorer leurs résultats.

1.2 Notions de biologie

Nous allons d'abord poser les bases en expliquant ce que sont les arbres. Ensuite nous verrons la façon dont ils poussent, ce qui peut être très utile en informatique pour simuler leur croissance et comprendre leur morphologie. Et enfin, nous présenterons les deux principaux éléments apportant des détails visuels aux arbres : l'écorce et les feuilles.

1.2.1 Qu'est-ce qu'un arbre ?

Sous cette question naïve repose en fait tout le problème qui nous intéresse. De plus, la réponse à cette question amènera par la suite d'autres questions, telles que : « Comment un arbre est-il constitué ? », « Qu'est-ce qui différencie un arbre d'un autre arbre ? », « La différence entre arbres et arbustes est-elle importante ? », etc.

1.2.1.1 Terminologie

Un arbre est une sorte de plante ligneuse ; c'est-à-dire qu'il est formé de branches qui sont plus ou moins « lignifiées » (de forme allongée) et dont la croissance est continue en longueur et en épaisseur [1]. C'est un végétal généralement constitué d'un tronc, de branches, de feuilles et de racines (voir Figure 5).

Le « fût » est la partie du tronc qui ne porte pas de branches. Il est situé entre les racines et la « couronne » qui est la zone de départ des branches maîtresses. Le tronc et les branches sont constitués de bois dont la partie extérieure, lisse ou rugueuse, est l'écorce. Les racines, principales et secondaires ainsi que leurs ramifications (les radicelles), sont invisibles la plupart du temps car sous la terre. Cependant, il n'est pas rare de voir apparaître hors de terre le début des racines principales, appelé « collet », qui est aux racines ce que la couronne est aux branches. Les plus jeunes branches n'ayant pas encore donné naissance à de nouvelles branches sont des « rameaux ». L'ensemble des branches et des rameaux s'appelle « ramure » ou « houppier » si l'on inclue le feuillage [106].

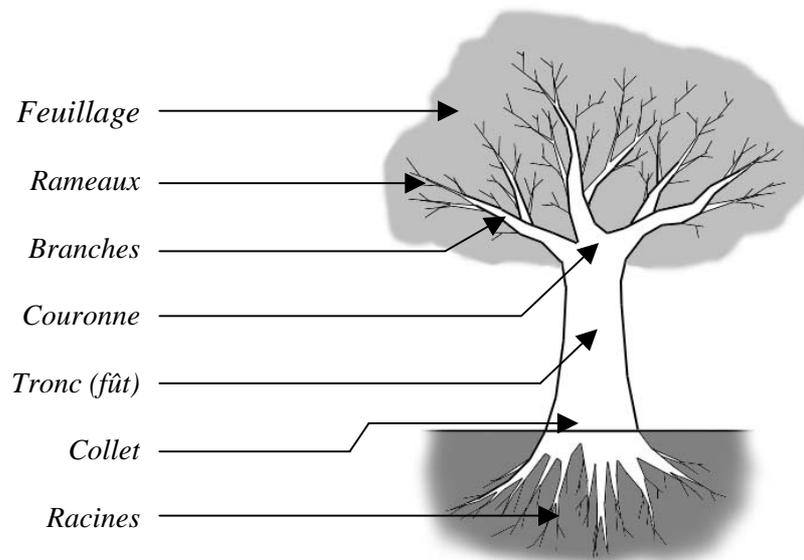


Figure 5 : Schéma d'un arbre feuillu quelconque. On retrouve toutefois les mêmes composantes dans l'autre catégorie d'arbres que sont les conifères (cf. paragraphe 1.2.1.2).

1.2.1.2 Classification

Il existe deux catégories d'arbres : les « feuillus » [1] [106] et les « conifères » [21]. Cette différence s'exprime d'un point de vue biologique par certaines caractéristiques de leurs graines. Les feuillus sont dits « angiospermes », c'est-à-dire que leurs graines (ovules) sont enfermées dans un espace clos (ovaire) ; alors que les conifères sont « gymnospermes », c'est-à-dire que leurs ovules sont nus. Mais d'un point de vue visuel – ce qui représente notre centre d'intérêt dans le cadre de cette thèse – notons surtout que les feuillus portent des feuilles (cf. paragraphe 1.2.4) dont le limbe est développé et plat tandis que les conifères ont des aiguilles ou des écailles.

Ces critères sont cependant très loin d'être les seuls à différencier les arbres les uns des autres. D'une espèce d'arbre à l'autre, on notera souvent de très importantes différences. Les plus visibles portent sur la forme générale, appelée « port », de l'arbre (élancé comme un peuplier ou ayant un houppier de forme arrondie comme un pommier ; immense comme un séquoia ou bien petit comme un olivier...) ; mais, de plus près, on note également que l'aspect de l'écorce est très varié (cf. paragraphe 1.2.3) et il en va de même pour la forme des feuilles (cf. paragraphe 1.2.4). De plus, rien qu'au cours du cycle

de vie d'un arbre, on peut noter de profonds changements dans l'aspect et la texture de son écorce et de ses feuilles, et dans une moindre mesure dans son port si l'on exclue la simple croissance. En fin de compte, il n'y a pas beaucoup plus de différences entre un conifère et un feuillu qu'entre deux feuillus ou deux conifères, tant la nature est diversifiée.

Ainsi, afin d'identifier chaque espèce d'arbre, les botanistes ont eu recours à la Taxonomie. Cette classification des arbres (ainsi que des autres espèces végétales et animales) est basée sur celle élaborée en 1735 par Carl von Linné dans son ouvrage « *Systema naturae* » puis perfectionnée pour les végétaux en 1753 dans « *Species plantarum* ». Linné a introduit le système de nomenclature binomiale qui consiste à regrouper les espèces vivantes (dans le cas qui nous concerne : les arbres) en une hiérarchie de catégories et de sous-catégories où les deux derniers niveaux (le genre et l'espèce) représentent le nom latin. Par exemple, « *Castanea sativa* » est le nom donné au « châtaigner commun ». Cependant, cette classification des arbres se fait selon certains critères qui, malgré tous les critères visuels mentionnés précédemment, portent encore une fois d'abord sur le mode de reproduction des arbres. Le genre dépendant de la structure des fleurs et des fruits, il n'est pas rare de trouver des arbres d'un même genre avec des écorces ou des feuilles visuellement assez différentes.

1.2.1.3 Arbre ou arbuste ?

Nous traitons indifféremment les arbres et les arbustes car la limite entre les deux est floue et c'est en général une simple question de dimensions qui les différencie ; un arbre étant supposé pouvoir dépasser trois mètres à l'âge adulte [107], même s'il existe de « grands arbustes » faisant exception à cette règle. Un autre critère parfois retenu pour différencier les arbres des arbustes, en plus de la taille, est la présence de tige (tronc) unique dans le cas des arbres, contrairement aux arbustes qui ont souvent leur couronne à même le sol, ce qui laisse penser qu'ils ont plusieurs tiges. Mais de nombreuses exceptions contredisent également parfois cette règle.

Quoi qu'il en soit, ces différences ne changent en rien la façon dont on peut représenter un arbre sur ordinateur car dans tous les cas, la partie visible d'un arbre sera composée d'une arborescence de branchages recouverts d'écorce et de feuilles (et éventuellement de fleurs ou de fruits).

1.2.2 La croissance des arbres

A l'origine de l'arbre, il y a la graine. De la graine pousse une tige qui sort de terre. Avec le temps, de l'eau, du soleil et un sol fertile, cette tige croît et s'épaissit par la poussée concentrique de nouvelles couches internes [43] jusqu'à devenir le « tronc », tandis que des bourgeons [44] apparaissent, donnant naissance aux feuilles et aux branches. Le tronc et les branches principales continuent de s'allonger chaque année et de nouveaux bourgeons donnent naissance à de nouveaux rameaux. Les branches s'allongent d'autant plus grâce aux bourgeons situés à leurs pointes, appelés « bourgeons apicaux ».

Au-delà de ces règles constantes, les arbres poussent de façon différente suivant leur espèce, bien sûr, qui leur fera prendre des formes bien particulières, mais aussi suivant leur environnement qui influera de façon significative sur leur développement.

1.2.2.1 La forme de l'arbre en fonction de son environnement

Les trois principaux facteurs naturels pouvant influencer sur la forme d'un arbre sont : le soleil, l'eau et le vent. Un bon ensoleillement et une bonne irrigation (pluie et proximité des nappes phréatiques) sont propices au développement de l'arbre. L'arbre dispose son feuillage de manière à capter un maximum de lumière ; ainsi, les branches sont légèrement orientées vers la lumière. Lorsqu'un arbre pousse au milieu des autres, il a tendance à pousser plus haut et être plus fin que s'il était en terrain découvert. Ceci est dû au fait qu'il recherche la lumière qui ne parvient pas jusqu'à lui à cause de ses voisins. De plus, l'arbre dirige ses racines pour pouvoir puiser le plus d'eau possible.

Les arbres poussant dans les lieux où souffle continuellement un vent prédominant ont des branches plus longues et plus saines sous le vent, de sorte que la couronne semble souffler avec lui [43] ; ce qui est principalement dû au fait que les bourgeons exposés au vent risquent de se détacher.

A ces facteurs naturels peuvent encore s'ajouter des agressions extérieures telles qu'une tempête qui briserait des branches et forcerait l'arbre à s'adapter afin de pouvoir à nouveau orienter les feuilles et les branches qui lui restent vers le soleil. Les animaux ou insectes parasites peuvent aussi endommager l'arbre.

1.2.2.2 Les branches

Les branches sont segmentées en « entre-nœuds » reliant deux nœuds consécutifs (voir Figure 6). C'est au niveau des nœuds que se situent les bourgeons et les feuilles.

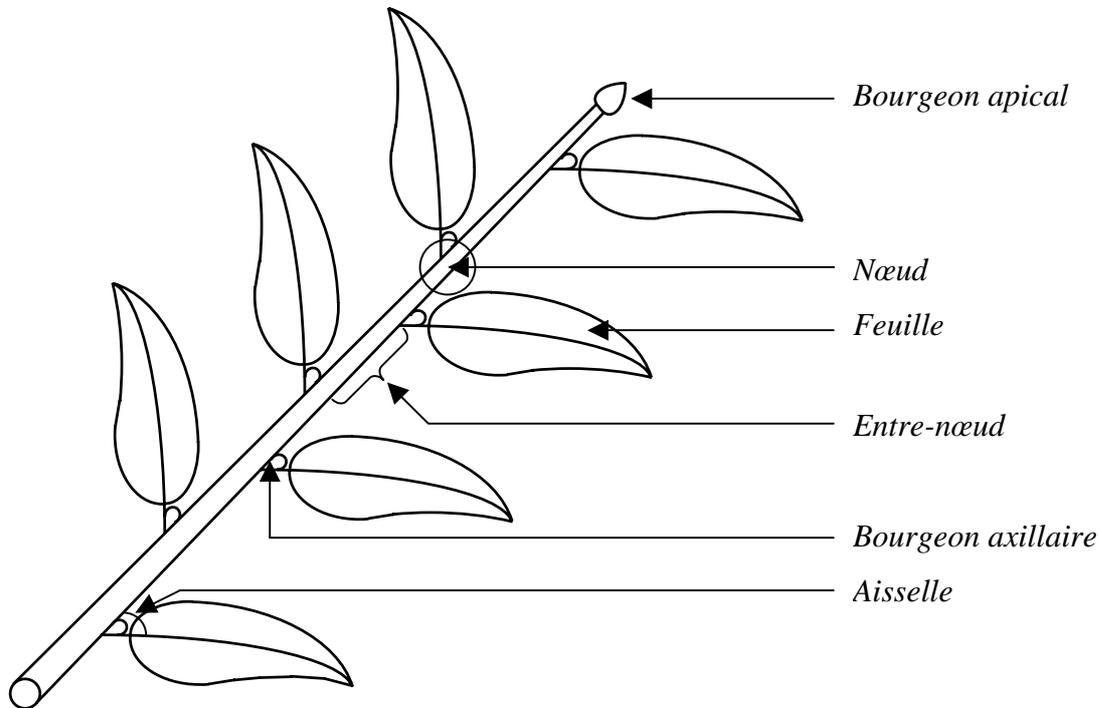


Figure 6 : Bourgeons et feuilles sur un rameau.

L'apparition de nouveaux rameaux se fait par les bourgeons qui peuvent rester dormants ou mourir [103]. Un bourgeon en bout de pousse est appelé bourgeon *terminal*, et un bourgeon à l'*aisselle* d'une feuille (angle entre la feuille et la tige) est appelé *axillaire* ou encore *latéral*.

Le développement des branches ne se fait pas de la même façon pour tous les arbres. Lorsqu'un bourgeon donne naissance à une nouvelle branche, celle-ci peut croître parallèlement à sa branche parente (verticalement, en général). On parle alors de croissance « orthotropique ». Si, au contraire, les nouvelles branches tendent à s'éloigner de leur branche parente, on dit que la croissance est « plagiotropique » (ou « dichotomique »). La Figure 7 schématise ce phénomène [103].

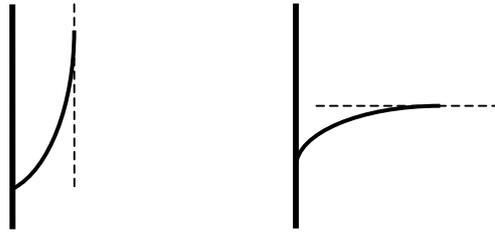


Figure 7 : Types de croissance des nouvelles branches. A gauche : croissance orthotropique. A droite : croissance plagiotropique.

Les nouvelles branches peuvent également naître alternativement d'un côté d'une branche puis du côté diamétralement opposé ou bien des deux côtés à la fois ou encore réparties aléatoirement sur la branche parente. De plus, les bourgeons ne donnent jamais naissances à de nouvelles branches la première année.

Les arbres pointus tels le sapin (dit de Noël) ont une croissance « monopodiale » ou encore à « prédominance apicale ». Cela signifie que leur sommet est la partie qui grandit le plus vite. Et, si le « bourgeon apical » du tronc (le bourgeon à la cime de l'arbre) venait à mourir, une des branches latérales du sapin se courberait vers le haut pour devenir le nouveau rameau apical [43]. Cette nouvelle croissance est dite « sympodiale » [21].

Pour de plus amples détails sur l'architecture des végétaux (leur structure et leur développement), le lecteur est encouragé à consulter l'ouvrage de Bouchon, Reffye et Barthélémy [20] qui présente tous les modes de croissance des arbres d'un point de vue informatique. Ce qu'il faut avant tout retenir à notre niveau, c'est la structure arborescente (squelette) articulée sur les nœuds.

1.2.3 L'écorce

La fonction principale de l'écorce est de protéger les arbres contre les « agressions » de leur environnement, de la même manière que la peau humaine protège notre chair. La partie visible de l'écorce est composée de couches de tissus morts. La croissance interne du bois pousse ces couches mortes à se fissurer ou à se craqueler puis parfois à tomber.

1.2.3.1 La structure de l'écorce

La partie de l'écorce qui nous intéresse est sa partie visible : son « épiderme », ou plus généralement le « rhytidome », c'est-à-dire la couche morte de « suber » (communément appelé « liège ») qui recouvre l'arbre (voir Figure 8). Le rhytidome est soit « annulaire », soit « écailleux » [126]. Son état accidenté est, en général, fonction du diamètre et donc également de l'âge de la branche. Ainsi le tronc qui est la plus grosse et la plus ancienne des parties de l'arbre sera plus rugueux que les rameaux. La rugosité de l'écorce dépend de la souplesse, de la persistance et de la caducité du rhytidome.

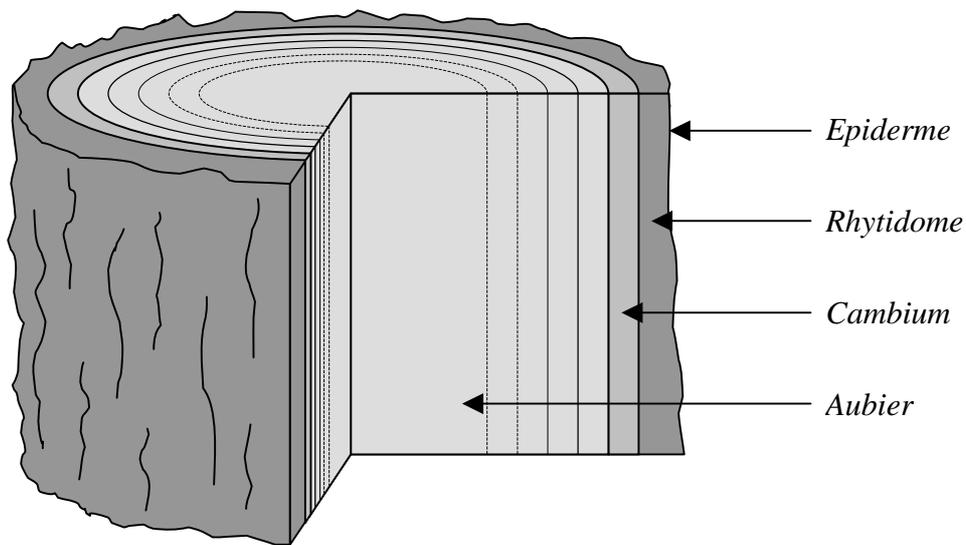


Figure 8 : Vue en coupe simplifiée d'un tronc ou d'une branche.

Cependant il existe un très grand nombre d'espèces d'écorces différentes. Lorsqu'un arbre grandit, l'épaississement concentrique qui part du cœur de la branche étire l'écorce et génère les craquelures sur la plupart des écorces d'arbres. Ainsi les arbres dont l'écorce est élastique seront moins craquelés que ceux qui ont une écorce rigide. On notera également que l'écorce des jeunes arbres est plus élastique que l'écorce d'arbres de la même espèce mais plus vieux. Ceci explique donc pourquoi l'écorce des arbres devient de plus en plus rugueuse avec l'âge. On peut d'ailleurs citer comme exemple d'évolution de l'écorce durant la vie de l'arbre, les larges craquelures noires qui apparaissent avec le temps sur les bouleaux. Ceci fait que les zones lisses de ce type d'écorce disparaissent progressivement au profit de zones plus rugueuses.

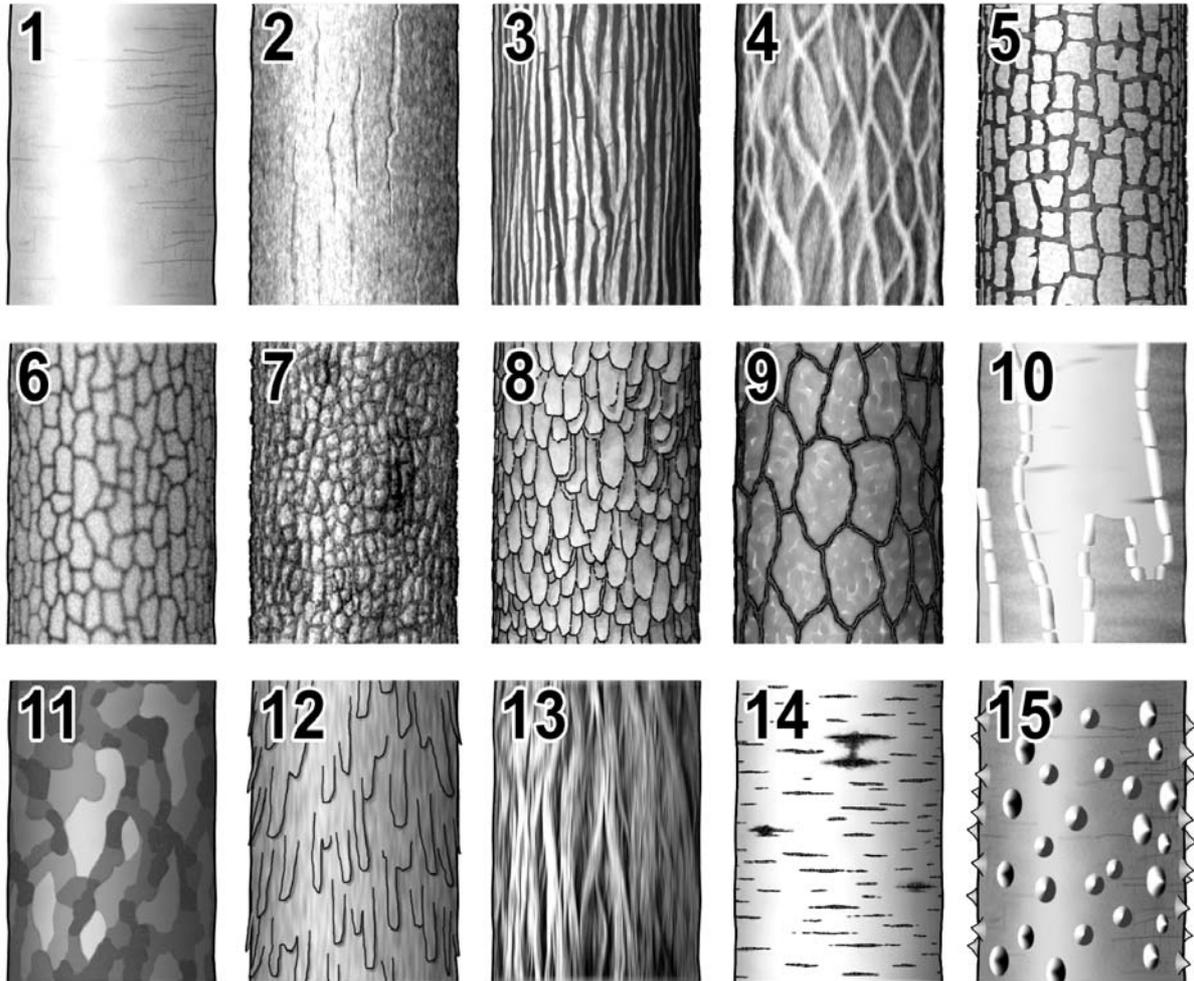
D'autre part, l'écorce de certains arbres est très fine. C'est le cas du « bouleau à papier » dont l'écorce éclate et s'enroule comme un papyrus avant de tomber ; ou encore celle des cerisiers et pruniers d'où sortent des protubérances allongées horizontalement aux endroits où l'écorce cède ; tandis que d'autres arbres comme les chênes ont l'écorce ravinée comme un sol aride. Enfin, la couleur des écorces change d'une variété à l'autre et suivant l'âge. Les couleurs les plus courantes sont : brun, gris, vert, beige, marron-rouge. Des espèces comme certains cerisiers et pruniers d'Asie ont même une écorce rougeâtre et « spéculaire » (qui réfléchit la lumière). Le lecteur est invité à découvrir la plupart des espèces d'écorces plus ou moins courantes dans le « Guide des écorces » [126].

1.2.3.2 Différencier visuellement les différents types d'écorces

Les arbres étant généralement classifiés suivant la nomenclature de Carl von Linné (cf. paragraphe 1.2.1.2), qui ne tient pas compte de l'aspect de l'écorce, beaucoup d'arbres de la même famille peuvent avoir des écorces complètement différentes.

Néanmoins, certains travaux dont ceux de Vaucher [126] ou bien le site Internet « Les Arbres » [144] ont tenté de dresser une liste des principales catégories d'écorces afin de pouvoir reconnaître les arbres par leur écorce, tout en gardant à l'esprit que l'écorce peut varier considérablement au sein d'une même famille et que l'aspect de nombreuses écorces combine plusieurs catégories. Bien que Vaucher considère la dureté des écorces comme un des critères de regroupement ou de distinction, nous nous concentrons seulement sur l'aspect visuel dans notre domaine qui est l'Infographie. La classification de la Figure 9 s'inspire de celle de Vaucher et, tout comme la sienne, n'est pas exhaustive et des catégories intermédiaires pourraient y être ajoutées bien qu'elle représente déjà une très large majorité des types d'écorces.

D'un point de vue purement informatique, sachant que la procédure habituelle pour synthétiser un phénomène consiste dans un premier temps à trouver empiriquement des règles simples qui permettent de le représenter dans le plus grand nombre possible de cas, nous verrons donc par la suite (au paragraphe 2.2) si cette diversité pose problème ou s'il est possible d'utiliser une telle classification ou un nombre encore plus réduit de catégories afin de développer des méthodes de rendu les plus générales possibles, quitte à traiter d'une façon différente les quelques cas particuliers restants.



Description visuelle des principales catégories d'écorce :

- 1 : lisse, légèrement granuleuse, parfois plissée ;
- 2 : fissures verticales peu profondes ;
- 3 : profonds sillons et crêtes verticaux ;
- 4 : profonde texture entrelacée ;
- 5 : très profondes craquelures formant des zones approximativement rectangulaires ;
- 6 : profondes craquelures formant des zones irrégulières ;
- 7 : profondes et très irrégulières craquelures et protubérances ;
- 8 : petites ou moyennes écailles ;
- 9 : larges écailles plates, souvent épaisses ;
- 10 : s'exfolie en fines pellicules qui s'enroulent ;
- 11 : s'exfolie en larges et minces morceaux laissant apparaître des zones plus claires ;
- 12 : s'exfolie en filaments ou lamelles ;
- 13 : fibreuse et verticale ;
- 14 : protubérances plus ou moins allongées ;
- 15 : lisse mais avec des épines.

Figure 9 : Classification des écorces suivant des critères visuels.

1.2.4 Les feuilles

1.2.4.1 Morphologie

Une feuille est la plupart du temps composée d'une queue, appelée « pétiole », qui se rattache au rameau et d'un limbe qui est le corps de la feuille. Le limbe d'une feuille qui ne possède pas de pétiole est dit « sessile ».

Forme du limbe

Il existe une multitude de caractéristiques morphologiques pour les limbes. Ils peuvent être « composés » (de plusieurs éléments séparés) ou non. Leur « marge » (bordure) peut être « entière » (lisse), dentée ou denticulée (petites dents). Leur forme peut être lobée (le contour forme des lobes), lancéolé (allongé), ovale, arrondi, etc. Les ouvrages de Godet [43] et de Rumelhart et Vidal [106] détaillent les principales caractéristiques morphologiques de ces feuilles et les illustrent pour un très grand nombre d'arbres. En outre, bien que les feuilles décrites ici sont celles des arbres feuillus, on peut plus ou moins assimiler les aiguilles des conifères à des feuilles allongées pointues et rigides.

Nervation

Le limbe est pourvu d'un réseau de nervures appelé « nervation » dont on distingue trois types (voir Figure 10) :

- la nervation « pennée », constituée d'une nervure médiane distincte d'où sont issues des nervures latérales de premier ordre qui rejoignent la marge du limbe et qui sont reliées entre elles par un réseau de nervures plus petites ;
- la nervation « parallèle », formée par des nervures de même épaisseur parallèles ou arquées qui vont de la base au sommet du limbe ;
- la nervation « palmée », dont les nervures principales forment un éventail et partent de la jonction entre le pétiole et le limbe.

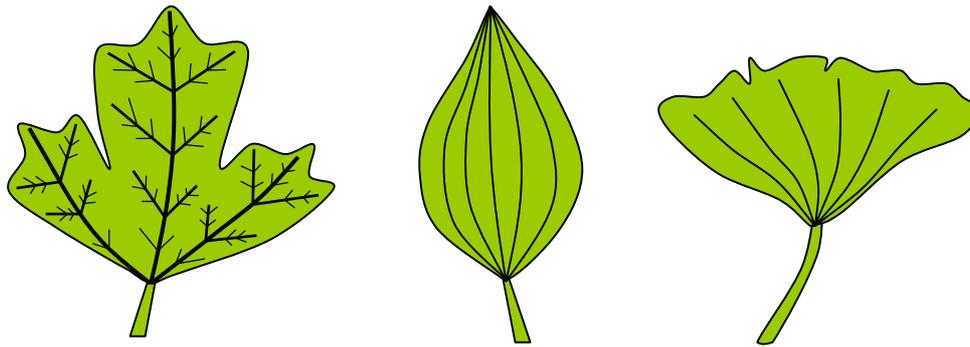


Figure 10 : Les différentes nervations. De gauche à droite : pennée, parallèle, palmée.

Disposition des feuilles sur la branche

Les feuilles peuvent se positionner sur la branche de différentes façons [43] (voir Figure 11). Elles peuvent être « alternes », « opposées », « décussées » (paires à angle droit autour de l'axe) ou « verticillées » (trois feuilles ou plus partant du même point).

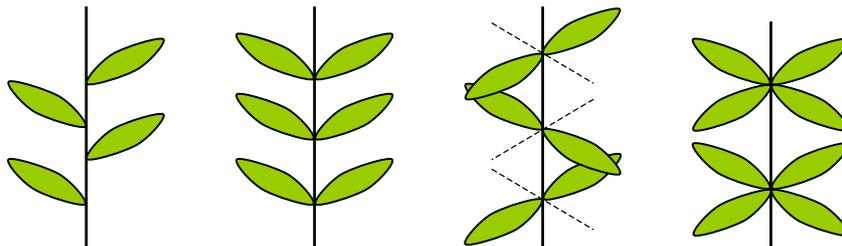


Figure 11 : Différentes dispositions des feuilles sur une branche. De gauche à droite : alternes, opposées, décussées, verticillées

1.2.4.2 Aspect

D'un point de vue visuel, les feuilles sont au moins aussi variées que les écorces comme le prouve la Figure 12 qui ne montre qu'une infime partie des formes de feuilles que l'on peut rencontrer dans la nature. Elles sont souvent spéculaires (brillantes) et foncées sur leur face supérieure alors qu'elles sont plutôt diffuses (mattes) et plus claires sur leur face inférieure. A noter que nous appellerons « supérieure » la face orientée vers le soleil (donc plutôt vers le haut) et « inférieure » la face opposée (donc orientée vers le sol). D'autre part, la couleur des feuilles dépend de l'espèce et de la saison.

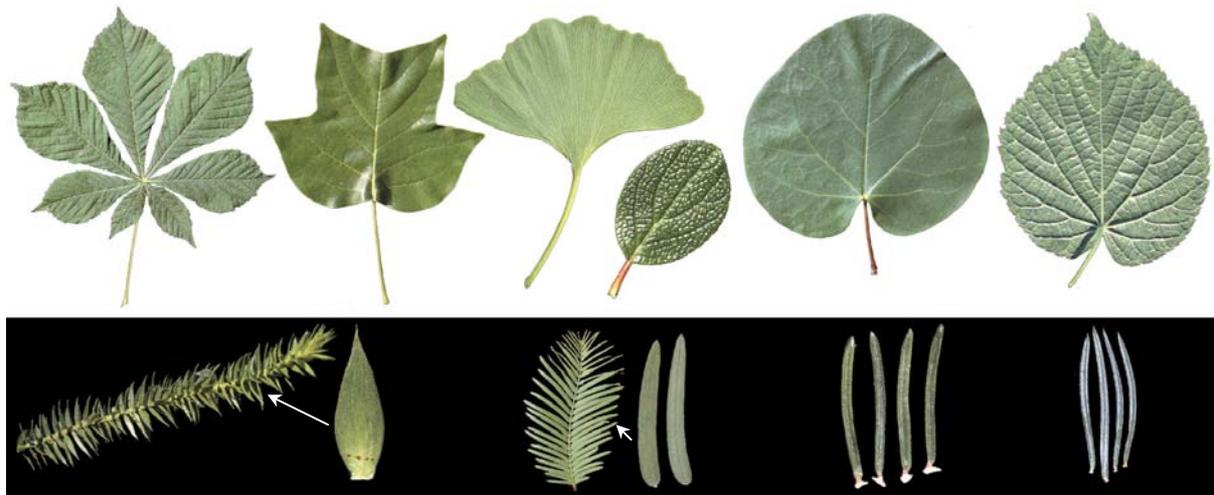


Figure 12 : Quelques exemples, très loin d'être exhaustifs, illustrant la grande diversité des de feuilles. Sur fond blanc : feuilles d'arbres « feuillus ». Sur fond noir : épinés de « conifères ».

De plus, étant donnée la fine épaisseur de la plupart des feuilles, elles sont souvent légèrement translucides. Un phénomène remarquable est la diffusion de la lumière à travers une feuille qui fait que très souvent, si l'on regarde une feuille à contre jour, parmi celles qui sont translucides, on constate qu'elle a l'air éclairée sur la face qui est pourtant non-éclairée, alors qu'avec une feuille opaque, la face non éclairée restera sombre.

1.2.5 Les autres éléments d'un arbre (fruits, fleurs et racines)

La diversité de la nature n'épargne ni les fleurs ni les fruits qui sont tous très différents en formes et en couleurs et qui représentent chacun un domaine de recherche à part entière. Le lecteur est invité à consulter des ouvrages sur les arbres [43] [106] [21] pour avoir un large aperçu de cette diversité.

En ce qui concerne les racines, nous nous y intéressons très peu car elles sont presque invisibles et les différents types de racines (pivotantes, mixtes, traçantes) [106] sont visuellement identiques au niveau du collet. De plus, on peu globalement traiter le collet de la même manière que la couronne.

1.3 Représentation d'arbres en infographie

La représentation des arbres en images de synthèse se décompose en plusieurs domaines distincts suivant le but recherché et la puissance de calcul disponible. Ainsi on distinguera les applications en temps réel privilégiant la rapidité d'exécution et les applications de rendu photo-réaliste s'attachant principalement à la qualité du rendu. On peut encore affiner les catégories de détails recherchés. Par exemple, pour le temps réel, on peut différencier les applications où l'arbre est le sujet principal et/ou en nombre suffisamment réduit pour qu'il soit bien détaillé, en opposition aux applications où l'arbre est secondaire et/ou en très grand nombre, auquel cas les sacrifices sur la qualité seront plus importants.

Il existe ainsi, comme l'a décrit Chaudy [22], deux types de représentations informatiques des arbres dont le choix dépend des considérations énoncées ci-dessus :

- la représentation « impressionniste » dont le but est d'obtenir un résultat visuel qui ressemble de façon satisfaisante à un arbre lorsqu'on l'observe d'assez loin, et qui est généralement utilisée dans les applications « temps réel » ;
- la représentation « structurelle » qui construit la structure détaillée de l'arbre pour le visualiser ultérieurement à l'aide d'une technique de rendu quelconque (selon les besoins : lancer de rayons afin d'obtenir un rendu précis mais parfois lent ou bien 3D accélérée matériellement pour un rendu parfois moins détaillé mais plus rapide).

Dans ce chapitre, nous détaillons beaucoup plus longuement la représentation structurelle des arbres puisque le but que nous recherchons en est le rendu le plus réaliste possible, et ceci même à courte distance, là où les détails de l'écorce, des feuilles et des branchages ressortent.

On peut enfin noter que certains travaux ont été menés sur le rendu non photo-réaliste des arbres [28] [38], ce qui permet de les insérer dans des scènes schématiques (notamment en architecture) ou d'adopter le style des dessins animés. Mais cette troisième catégorie de représentation s'éloigne du domaine qui nous intéresse et découle en fait bien souvent d'un rendu particulier (forçant sur le dessin des contours plus que sur les détails) effectué sur la base réaliste d'une représentation structurelle.

1.3.1 Représentation « impressionniste »

1.3.1.1 Représentation par « panneaux » (images/photos 2D)

L'idée de base de cette méthode est de représenter un arbre dans une scène au moyen d'un simple polygone rectangulaire sur lequel on plaque une image d'arbre, avec gestion de la transparence (à l'aide d'un masque par exemple). Son but est donc, avant tout, la rapidité d'affichage. L'image (texture) utilisée est appelée « panneau » ou « *billboard* » et peut être une photographie afin de donner un semblant de réalisme au résultat obtenu ou bien un arbre synthétisé avec une des meilleures méthodes structurales (cf. paragraphe 1.3.2). Mais il est évident que cette méthode comporte plusieurs inconvénients :

- **On doit stocker autant de textures que l'on veut d'arbres visuellement différents.** Si l'on souhaite représenter une forêt avec une multitude d'arbres, même s'ils sont de la même espèce, il faudrait en théorie une texture par arbre car chaque arbre est censé être unique. Ce n'est cependant pas un problème très gênant car on utilisera de toutes façons cette méthode dans des cas où l'on ne porte pas un intérêt capital aux arbres et l'on peut donc se permettre de répéter plusieurs fois un même modèle d'arbre. Toutefois il en résultera un effet de répétition si l'on souhaite limiter l'occupation de la mémoire.
- **On peut très difficilement gérer l'éclairage dans le feuillage.** C'est un problème un peu plus gênant car si l'on prenait par exemple un arbre photographié un jour de plein soleil avec un éclairage venant de gauche et que l'on souhaitait l'inclure dans une scène où l'éclairage viendrait de droite ou bien où le ciel serait nuageux, il n'y aurait plus de cohérence entre les arbres et le reste de la scène. Certains travaux tentent d'apporter une solution à ce problème comme par exemple ceux de Qin et al. [100].
- **Les arbres sont plats.** Le problème majeur des « panneaux » est bien sûr la platitude des images employées. Un problème de taille intervient lorsqu'on veut déplacer le point d'observation (la caméra) dans la scène. D'une part cette méthode suppose que tous les arbres sont affichés face à l'observateur (plaqués perpendiculairement à l'axe de vision) afin de ne pas voir d'arbres sur leur tranche ; donc si la caméra effectue une rotation, on risque de voir les arbres effectuer eux aussi une rotation. D'autre part, il faut conserver la contrainte de perpendicularité des arbres par rapport au sol. Par conséquent, un survol des arbres, dans un simulateur de vol par exemple, ferait que l'on se rendrait compte de

la platitude des arbres si l'on se mettait à regarder vers le sol. En effet, comme les arbres doivent rester perpendiculaires au sol, si la direction du regard est, elle aussi, pratiquement perpendiculaire au sol, elle sera par conséquent parallèle à l'arbre, ce qui nous ferait voir les arbres sur la tranche supérieure du « panneau ».

Pour remédier en partie à l'impossibilité assez handicapante de se déplacer librement dans la scène, on peut plaquer deux vues de l'arbre en croix avec un angle de 90° d'écart (voir Figure 13). Cette méthode a très souvent été utilisée dans les jeux vidéo à l'époque où les cartes graphiques étaient moins performantes et elle continue toujours à l'être dans les jeux où les arbres sont secondaires ou assez petits/éloignés pour ne pas nécessiter plus de précision.

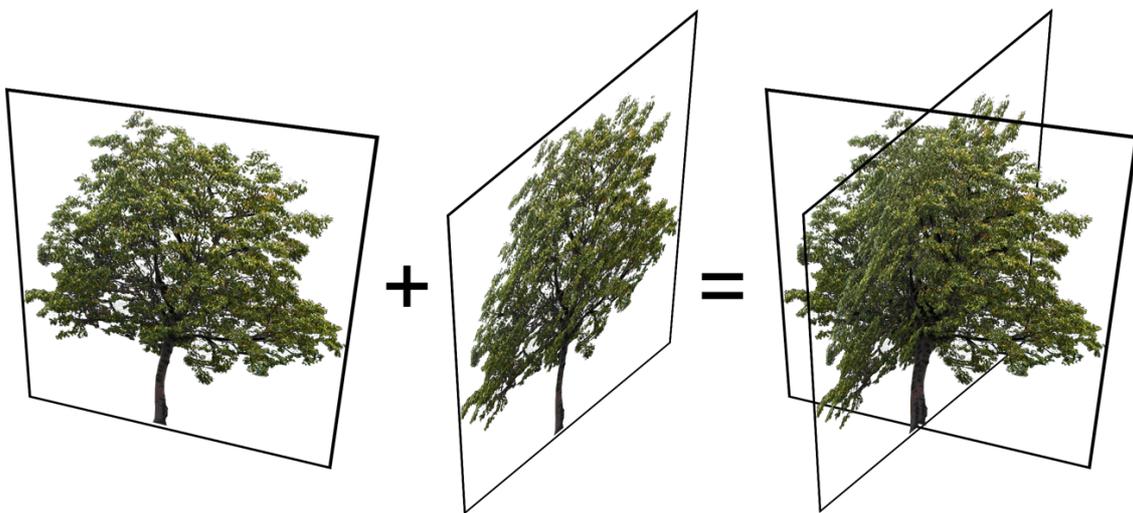


Figure 13: *Panneaux en croix pour donner une impression de volume et permettre de se déplacer autour de l'arbre.*

Cependant cela ne corrige pas le cas où l'on observe l'arbre de dessus. On peut alors parfois rencontrer une troisième texture représentant l'arbre vu de dessus ; mais, cette texture est souvent dessinée car, en pratique, il est trop compliqué de prendre une photographie d'un arbre vu de dessus et d'épurer la photo afin d'exclure le sol (en le remplaçant par des pixels transparents) pour ne garder que l'arbre.

On peut également recourir aux « imposteurs » [27] qui sont des ensembles de textures choisies selon le point de vue et/ou la direction de la lumière. De nombreux travaux ont été menés pour tenter de solutionner ces difficultés rencontrées lors du

déplacement de la caméra et/ou la modification de l'éclairage dans une scène en proposant de représenter l'arbre non pas par seulement un, deux ou trois panneaux mais par plusieurs panneaux parallèles suivant les trois axes de l'espace [19] [59] ou par des ensembles de polygones, comme les méthodes décrites au paragraphe suivant, mais dont la précision s'adapte selon la distance par rapport au point de vue en affichant plus de détails sur le côté de l'arbre qui fait face à la caméra [104]. Une autre méthode adaptée à la distance a été proposée par Meyer et al. [85] pour représenter les arbres par un ensemble de voxels (petits cubes discrétisant le volume occupé par l'arbre) ayant des propriétés de couleur et d'éclairage.

On retrouvera la plupart du temps ces représentations dans des applications dites « temps réel » qui nécessitent une grande rapidité de traitement que l'on obtient au détriment de la précision. Cependant il n'est pas envisageable d'observer de près les arbres ainsi générés car on verrait alors inévitablement le manque de relief et les pixels grossis si l'on n'a pas prévu une texture 2D suffisamment précise ou un ensemble de voxels discrétisé suffisamment finement. On réserve donc principalement ces modèles pour des jeux vidéo ou certains simulateurs qui doivent être rapides et pour lesquels on peut être indulgent quant à la qualité du rendu, surtout si l'on reste éloigné des arbres.

1.3.1.2 La solution actuelle retenue en général dans les applications temps-réel

De nos jours, à moins d'avoir une énorme quantité d'arbres à afficher, on trouve dans les applications en temps-réel – et en particulier les jeux vidéo – des arbres plus détaillés que les simples panneaux en croix (cf. paragraphe précédent), comme l'illustre la Figure 14. Ces arbres sont à la limite de la modélisation structurelle (cf. paragraphe 1.3.2) mais sont le plus souvent modélisés manuellement, notamment au niveau des branches, et comportent un grand nombre de textures, souvent dessinées à la main, afin de simuler le feuillage avec un coût relativement faible en polygones.

Ainsi, au lieu d'avoir une seule texture représentant l'arbre entier, ce dernier est composé de plusieurs polygones pour former le tronc, les branches et le feuillage. Le tronc et les branches principales sont modélisées sous forme de volumes à base de polygones tandis que les plus petites branches et les feuilles dont le volume est négligeable sont regroupées et représentées par des textures avec transparence (composante alpha), à la

manière du « *billboarding* par micro-facettes » [137] mais sans la nécessité que les micro-facettes soient dépendantes du point de vue.



Figure 14 : Exemples d'arbres extraits du monde des jeux vidéo. En haut : « *Medal of Honor : Allied Assault* »(jeu, ©2002 Electronic Arts). En bas : « *CodeCreatures Benchmark* » (logiciel principalement destiné aux joueurs pour tester les performances de leur PC, ©2002 CodeCult). Les branches principales ont du volume alors que les feuilles et les petites branches sont remplacées par des textures 2D transparentes.

De plus, pour des raisons d'économie de mémoire, les textures représentant le feuillage et les petites branches, ainsi que celles représentant l'écorce, sont souvent dessinées en basse résolution. Ceci permet dans les deux cas de justifier qu'elles soient parfois dessinées à la main et, parfois dans le second cas (écorce), de façon à « boucler » pour être juxtaposées le long des branches sans discontinuité visible.

1.3.1.3 Systèmes de particules

Reeves et Blau [102] ont adapté aux végétaux le système de particules défini par Reeves [101], utilisé à l'origine pour simuler des nuages, du feu et de l'eau. Une particule est un petit élément (généralement sphérique) qui est constitué des attributs suivants :

position dans l'espace, taille, vecteur vitesse, couleur, opacité, âge, durée de vie, fonction à exécuter en fin de vie, etc. Grâce à ces informations, on peut définir une trajectoire pour un ensemble de particules et connaître à chaque instant l'état d'une particule (pas encore née, vivante et dans un certain état, morte). Tout ceci représente un système de particules.

Ainsi, les branches d'un arbre peuvent, chacune, être représentées par le trajet d'une particule structurée de façon à naître au début de la branche, et mourir au bout en créant éventuellement des feuilles (qui peuvent être, elles aussi, des particules) et d'autres branches. Ce qu'on appelle ici « branche » est plus souvent un « segment de branche », avec la possibilité qu'un nouveau segment de branche, dont la trajectoire sera sensiblement la même que celle du segment actuel, soit créé à la mort de ce dernier afin de continuer la branche qui sera donc formée d'une suite de segments de branche.

Si ce type de modélisation est classé dans la catégorie « impressionniste », c'est parce que la détermination de l'illumination de chaque particule se fait de manière « probabiliste » – c'est-à-dire que l'on considère qu'une particule proche du centre de l'arbre aura moins de chances d'être éclairée qu'une particule située sur la périphérie de l'arbre et de surcroît du côté de la source lumineuse (de plus on ne prend pas en compte les arbres voisins qui peuvent eux aussi faire de l'ombre) – et parce l'utilisation de sphères pour représenter les particules ne peut que donner l'impression qu'elles forment un arbre pas permettre d'atteindre un rendu vraiment réaliste. Les résultats sont néanmoins très bons vus de loin comme le montre la Figure 15, même si le manque de détails deviendrait apparent en se rapprochant des arbres.

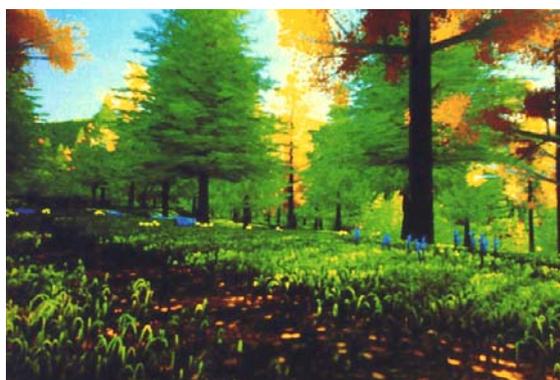


Figure 15 : Image extraite du film « *The adventures of André and Wally B.* » réalisé par W. T. Reeves et R. Blau.

1.3.1.4 Autres méthodes

Il existe encore d'autres méthodes qui permettent d'obtenir un résultat ressemblant à des arbres, comme celle de Gardner [41] qui consiste à représenter le feuillage d'un arbre comme un ellipsoïde dans lequel on contrôle la densité de matière (feuilles) grâce à une fonction à base de sinusoides. Mais, bien qu'elle permette d'obtenir des nuages assez réalistes et qu'elle soit assez rapide et d'une relative simplicité à mettre en œuvre, elle n'est pas adaptée à la modélisation d'arbres qui paraîtront réalistes de près. Toutefois, en ne simulant que des groupes de feuilles à l'aide de ces ellipsoïdes, que l'on disposerait sur des branches modélisées de façon « structurelle » (voir paragraphe suivant), on s'approcherait de la méthode [137] décrite au paragraphe 1.3.1.2.

1.3.2 Modélisation « structurelle » (du squelette de l'arbre)

La modélisation « structurelle » d'un arbre consiste à créer son squelette (structure en « fil de fer » des branchages comme le montre la Figure 16) puis, dans un second temps, à donner du volume et une texture aux branches et à rajouter des feuilles (et éventuellement des fleurs, des fruits et d'autres détails), sachant que le squelette devra comporter des informations relatives à la position des feuilles. Tous les modèles sont plus ou moins basés sur la récursivité, car c'est un phénomène très souvent rencontré dans la nature comme l'a fait remarquer Mandelbrot [80] et en particulier dans les arbres où chaque branche (jusqu'aux rameaux exclus) est composée de branches plus petites ayant globalement la même structure que la branche parente.

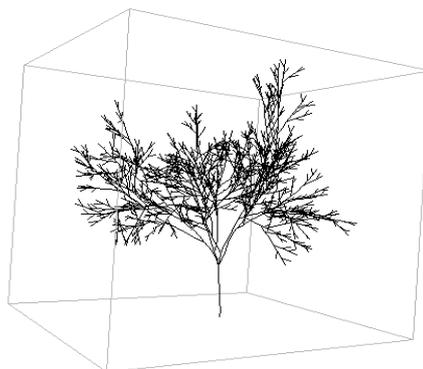


Figure 16 : squelette d'un arbre représenté en « fils de fer ».

Nous traitons ici des méthodes qui ont été conçues pour modéliser la structure des arbres virtuels et nous verrons les phases d'ajout de volume, de texture d'écorce, etc. dans les chapitres suivants.

1.3.2.1 Les débuts

Les premiers travaux réalisés entre autres par Honda [56] et Aono et Kunii [3] se contentaient de modéliser le squelette d'arbres « parfaits », c'est-à-dire des arbres qui ont récursivement la même structure (par exemple : un arbre dont chaque branche se sépare en deux pour donner naissance à deux branches qui se sépareront, elles aussi, en deux à l'étape suivante, et ainsi de suite), et ceci à l'aide de paramètres permettant de régler la taille et l'angle de branchement de chaque branche. La visualisation se faisait à l'époque uniquement à l'aide de segments de droites de taille variable.

Honda [56] a décrit la représentation des arbres dont la structure des branches est soit « dichotomique » soit « monopodiale » (cf. 1.2.2.2). Ses travaux furent repris par Aono et Kunii [3] qui ont mis en oeuvre une sorte de « L-système » (voir paragraphe suivant) avec des formules mathématiques simples dont les paramètres sont déduits de l'observation des arbres réels. Ces formules permettent, entre autres, de déterminer l'angle des rameaux naissant sur une branche. Ces modèles supportent des ramifications de type binaire ou ternaire, ce qui permet de représenter la plupart des arbres.

1.3.2.2 Les modèles à base de « L-systèmes »

Les « L-systèmes » [73] [74] furent introduits par Lindenmayer en 1968 et sont une extension des grammaires de réécriture pour les structures arborescentes en utilisant des crochets ('[' et ']') pour spécifier les branches (voir l'exemple de la Figure 17). Se basant sur la notion de récursivité inhérente à la nature – et en particulier aux arbres – et, partant du constat qu'on ne peut bien déterminer l'état d'un arbre que si l'on connaît la façon dont il s'est développé [122], les grammaires semblent bien adaptées car il semble naturel de partir d'une « graine » (l'axiome de la grammaire) et de simuler son développement en la remplaçant par une tige formée d'entre-nœuds et de bourgeons qui seront remplacés à l'étape suivante par de nouvelles branches tandis que les entre-nœuds se développeront (grossissement, allongement, etc.).

Smith [110] fut un des premiers à utiliser ces grammaires de graphes pour représenter des structures plus ou moins « fractales » [80] telles que les arbres, c'est-à-dire des structures se répétant récursivement tout en présentant de plus en plus de détails au fur et à mesure qu'on augmente la profondeur de génération. C'est évidemment le cas des branches d'un arbre que l'on peut représenter grossièrement par son tronc et ses quelques branches principales mais qui comporte en fait des branches de plus en plus petites reprenant à peu près toutes le même format de ramification. Ces grammaires, appelées « graftales » bien qu'aujourd'hui on emploie plutôt le terme original de « L-systèmes » en ce qui concerne les arbres, sont donc constituées de règles de réécriture qui définissent l'arbre à différentes échelles. Par la suite, Oppenheimer [87] a réutilisé les fractales en intégrant une notion d'aléatoire qui leur manquait jusqu'alors.

Le principe de base (dont, par exemple, un tutorial a été écrit par Jones et Kaandorp [62]), illustré par la Figure 17, est donc de représenter chaque élément de l'arbre par des éléments du vocabulaire terminal ou non de la grammaire. Par exemple, des caractères non-terminaux pourront être assignés aux segments de branches et aux bourgeons qui se réécriront respectivement en segment de branche plus long et en nouveau rameau à l'étape de croissance suivante (une année).

Dans cet exemple, la chaîne générée à la troisième étape de récursivité est donc : « SS [S[A]S[A]A] SS [S[A]S[A]A] S[A]S[A]A ».

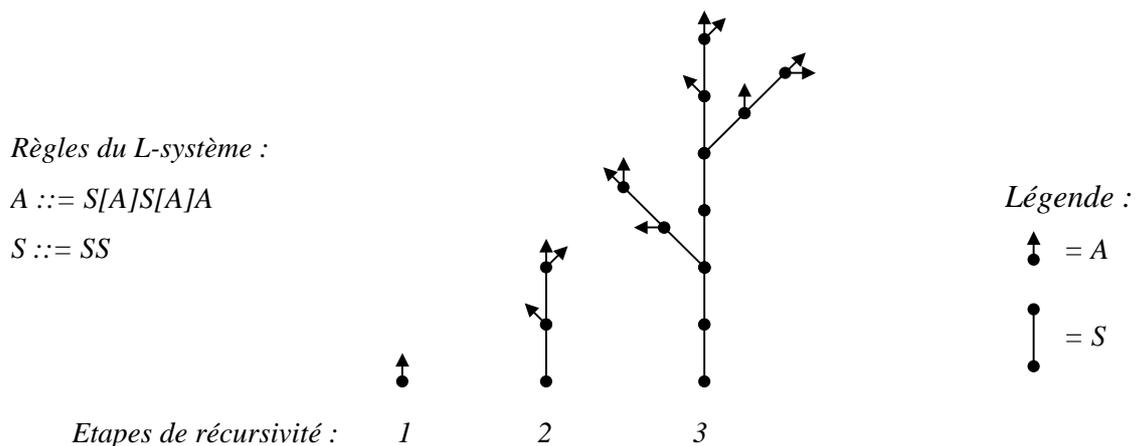


Figure 17 : Exemple simple de L-système dont l'axiome est A. On peut assimiler A à un bourgeon et S à un segment d'entre-nœud.

Prusinkiewicz et Hanan se sont alors associés à Lindenmayer [98] pour étendre le modèle de ce dernier. Ils ont rajouté des caractères de contrôle au vocabulaire de la grammaire afin de pouvoir piloter une tortue, similaire à celle du langage LOGO, destinée à dessiner les arbres à partir des chaînes générées. Les caractères '[' et ']' servent donc à empiler et dépiler l'état de la tortue le temps de traiter la branche définie entre crochets.

Parmi les caractères introduits, il y en a qui permettent d'effectuer une rotation de la tortue d'un angle prédéfini suivant les 3 axes de son repère local ou bien faire un demi-tour ('+', '-', '&', '^', '/', '\', et '|'). On peut ainsi définir l'angle de raccordement des nouvelles branches. D'autres caractères peuvent être utilisés pour gérer le diamètre ou encore la couleur des segments de branches. Par exemple, le symbole '!' réduit par un facteur prédéfini (habituellement par 2) le diamètre des futurs segments générés.

D'autres améliorations ont été apportées à ce modèle par Prusinkiewicz et al. [61] [84] [99] pour gérer encore plus de notions physiques et botaniques. Des conditions ont été intégrées à la grammaire pour plus de souplesse et gérer plus de cas. D'autre part, il est possible de contraindre la forme de l'arbre pour simuler des « topiaires » (arbustes taillés)¹. Ainsi, de par leur possibilité de représentation assez fidèle de la croissance des arbres, les L-systèmes sont aujourd'hui très utilisés.

Enfin, comme il n'est pas toujours évident de définir la grammaire représentant une espèce d'arbre, Shlyakhter et al. [112] ont proposé une méthode qui retrouve le volume englobant d'un arbre à partir d'un ensemble réduit de photos de cet arbre vu sous différents angles. Ce volume permet alors de contraindre un L-système comme un topiaire. Ainsi, on peut se contenter de ne définir que grossièrement la grammaire de l'arbre.

1.3.2.3 Un modèle combinatoire

Viennot et al. [127] utilisent ce qu'ils appellent une « matrice de ramification ». Celle-ci permet de déterminer la probabilité pour qu'un segment de branche d'ordre k se décompose en deux sous-branches d'ordres j et k avec $j < k$ ou bien toutes les deux d'ordre $k-1$, sachant que l'ordre des segments terminaux est 1. En d'autres termes, cela permet de

¹ Lintermann et Deussen [76], ainsi que Weber et Penn [132] ont également géré les topiaires.

représenter des grosses branches à partir desquelles partent de plus petites branches ou qui prennent fin pour donner naissance à de plus petites branches (cela peut être le cas du tronc par exemple). Cette matrice est triangulaire et la première ligne est facultative car les nœuds d'ordre 1 sont terminaux et ne peuvent donc pas se décomposer. De plus, la somme des probabilités de chaque ligne de la matrice doit être égale à 1. La Figure 18 montre un exemple de représentation d'arbre par cette méthode.

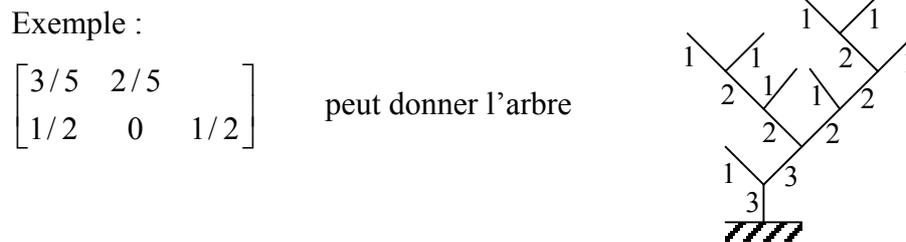


Figure 18 : Exemple de représentation d'arbre avec une matrice de ramification.

Dans cet exemple, la première ligne (d'ordre 2) indique qu'un segment d'ordre 2 a trois chances sur cinq de se poursuivre en générant un segment d'ordre 1 et a deux chance sur cinq de se décomposer en deux segments d'ordre inférieur (donc d'ordre 1, ici). Pour la seconde ligne de cet exemple, on constate qu'un segment d'ordre 3 ne peut pas se poursuivre tout en générant un segment d'ordre 2 (probabilité nulle).

Le problème majeur de cette méthode est cependant qu'elle ne permet de générer que des arbres binaires, ce qui représente toutefois un nombre déjà conséquent d'arbres.

1.3.2.4 Un modèle basé sur la géométrie

Weber et Penn [132] ont élaboré un modèle basé sur un nombre relativement important de paramètres précis ne nécessitant pas de connaissances particulières en Botanique. Ainsi, l'utilisateur peut régler des paramètres tels que la taille du fût de l'arbre, les dimensions des branches, leurs angles, le nombre de troncs de cônes composant chaque branche, le nombre de sous-branches, le nombre de feuilles, le niveau de récursivité, etc. D'ailleurs, l'utilisateur a la possibilité de régler tous ces paramètres pour chaque niveau de récursion. D'autres paramètres intéressants ont été prévus pour modifier l'aspect des troncs de cônes formant les branches de l'arbre ; ce qui permet d'évaser la base d'un tronc pour

représenter le collet de l'arbre. Alors, pour peu que l'on règle les bons paramètres, on peut obtenir des arbres relativement fidèles à ce que l'on cherche.

1.3.2.5 Les modèles basés sur des connaissances botaniques

Puisque le but de la plupart des modèles structurels est de représenter « fidèlement » la structure d'un arbre, beaucoup de modèles utilisent des notions de botanique prenant en compte les éléments internes et externes de l'arbre et participant à son développement.

Simulation de l'activité des bourgeons

On peut citer l'exemple de Reffye et al. [103] qui prennent en compte toutes les notions exposées au paragraphe 1.2.2. Mais le point principal de leur article est le modèle de croissance lui-même qui se fait en simulant l'activité des bourgeons (vivant et actif ; vivant et en pause ; mort) au cours du temps. L'algorithme général est le suivant :

```
POUR CHAQUE unité de temps (année) FAIRE
  POUR CHAQUE bourgeon encore vivant FAIRE
    Déterminer aléatoirement le nouvel état du bourgeon
    SI le bourgeon est vivant et actif ALORS
      Créer entre-nœud (EN) à la place du bourgeon
      Créer bourgeon apical au bout de l'EN
      POUR CHAQUE bourgeon possible sur EN FAIRE
        SI ramification ALORS
          Créer bourgeon axillaire
        FIN SI
      FIN POUR
    FIN SI
  FIN POUR
FIN POUR
```

Influence de la lumière

Beneš [8] a étudié l'influence de la lumière sur l'orientation des branches (« phototropisme » et « héliotropisme ») et a constaté que les bourgeons ont plus tendance

à rester en pause (voir la méthode précédente de Reffye et al. [103]) s'ils manquent de lumière. Měch et Prusinkiewicz [84] ont aussi intégré une gestion de la lumière à leur méthode à base de L-systèmes ; notamment la compétition de plusieurs arbres pour gagner de la lumière où l'on voit que les branches qui poussent vers l'arbre voisin sont atrophiées lorsque deux arbres sont très proches. Ils ont pour l'occasion baptisé leur extension des L-systèmes en « *Open L-systems* ».

Holton [55] a proposé une méthode pour permettre à la lumière, mais aussi à la gravité et au vent (également traité par Stam [117] dans le cas de turbulences), d'influencer le port de l'arbre. Enfin, on peut citer Soler et al. [115] qui ont repris les travaux de Reffye et al. [103] en y ajoutant une gestion avancée de l'éclairage.

1.3.2.6 Un modèle interactif

Lintermann et Deussen [75] [76] [77] ont développé une interface qui permet de définir de façon interactive des structures végétales très complexes. La Figure 19 illustre quelques exemples des bons résultats qu'ils obtiennent. Cependant, ils admettent que seul un utilisateur expérimenté peut arriver à un tel résultat (mais ceci en seulement quelques heures), ce qui est de toutes façons le cas avec la plupart, voire la totalité, des méthodes de modélisation d'arbre qui nécessitent une phase d'apprentissage avant de les maîtriser.



Figure 19 : Deux exemples d'arbres obtenus par Deussen et Lintermann.

Pour éditer leurs végétaux à l'aide de leur logiciel nommé « *Xfrog* », ils disposent d'une panoplie compacte d'icônes représentant des opérations qu'ils peuvent enchaîner à l'aide d'un graphe. A chacune des occurrences de ces icônes dans le graphe est associée une fenêtre de configuration des paramètres liés à l'icône. Certains de ces paramètres ne sont d'ailleurs pas sans rappeler ceux de Weber et Penn [132].

Par exemple, ils disposent entre autres d'un objet « corne » qui est en fait un cône déformable. C'est l'objet de base pour la construction des branches. Ils disposent également d'une opération « arbre » qui reproduit une structure sur un niveau de récursivité. Grâce à ces deux icônes (on peut même se contenter uniquement de l'icône « arbre » puisque par défaut, cette opération utilise une « corne » comme objet à répéter), on peut donc créer une branche (ou tronc), puis l'appliquer un certain nombre de fois (paramétrable) sur le tronc. On obtient alors un tronc avec les branches principales. Si l'on emploie cette opération une seconde fois, on va rajouter des rameaux aux branches principales. Enfin, si au bout du graphe, on rajoute l'icône « feuille », on peut alors obtenir avec de l'expérience un arbre ressemblant à ceux de la Figure 19.

De nos jours, la plupart des générateurs d'arbres sont interactifs. Même ceux qui manipulent des L-systèmes nécessitent un réglage interactif des paramètres pour gérer toutes les extensions ajoutées depuis les débuts de L-systèmes (comme dans le logiciel LStudio de l'Université de Calgary). Parmi les plus célèbres, on peut citer *AMAP* développé par le CIRAD ou encore le logiciel *OnyxTREE* de Onyx Computing, bien que ce dernier ne permette que de modifier des modèles d'arbres prédéfinis.

1.3.3 Du volume pour les branches

Parmi les méthodes de modélisation du squelette décrites précédemment, certaines se contentent de donner la structure principale de l'arbre sous forme de segments de branches auxquels il faut donner du volume alors que d'autres (Kawaguchi [65], Weber et Penn [132], Deussen et al. [75] [76] [77], etc.) génèrent directement le volume des branches sans forcément passer par le squelette. Pour ces dernières, il est cependant très souvent possible de retrouver le squelette sous-jacent afin de se retrouver dans le premier cas de figure. Il est en effet préférable de donner soi-même ce volume aux branches afin de pouvoir, entre autres, y appliquer une texture d'écorce comme on l'entend.

Les différentes façons de donner du volume au squelette de ces branches, selon les besoins et la puissance de calcul disponible, sont (du plus léger au plus lourd en terme de calculs) : les cylindres et les troncs de cônes, les cylindres généralisés, et enfin les surfaces implicites.

1.3.3.1 Cylindres

Partant du constat simple que les branches sont de forme plus ou moins cylindrique, la première forme de base qui a été employée pour donner du volume à chaque segment de branche fut le cylindre. Ainsi, en donnant un rayon à chaque segment du squelette de l'arbre, on a pu obtenir des résultats visuellement plus représentatifs que de simples « fils de fer ». Toutefois, cette méthode simple présente deux inconvénients majeurs (et quelques autres encore que nous verrons plus tard) illustrés par la Figure 20 :

- Les branches ayant un diamètre qui décroît de leur base jusqu'à leur extrémité, la représentation par cylindres, dont le rayon est constant le long de chaque segment de branche, engendre une discontinuité à la jonction de deux cylindres de rayons différents ;
- Un autre type de discontinuité apparaît aux jonctions si les cylindres ne sont pas colinéaires, ce qui est le cas le plus fréquent, car les arbres n'ont jamais des branches parfaitement droites.

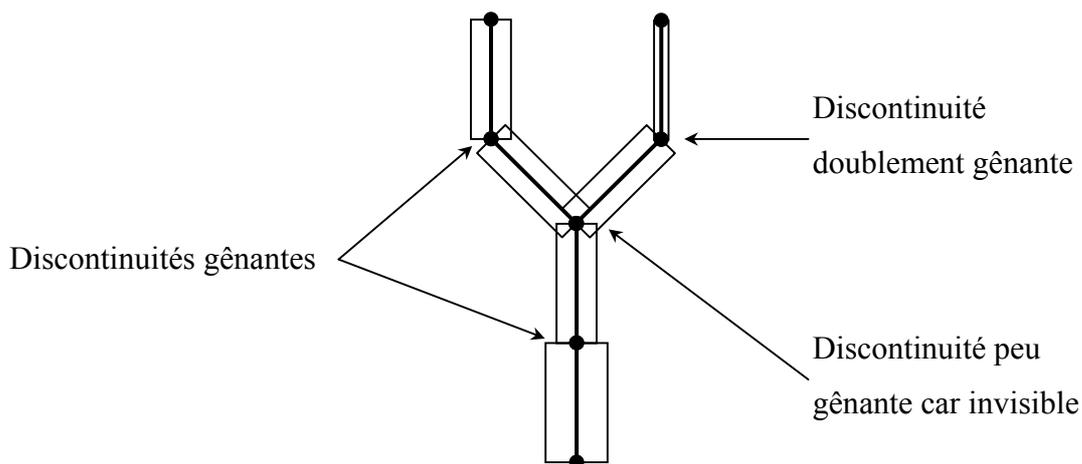


Figure 20 : Schéma d'un arbre représenté avec des cylindres.

1.3.3.2 Troncs de cônes

Afin de résoudre le premier problème rencontré avec les cylindres (discontinuité au raccord de deux cylindres de diamètres différents), on a alors utilisé des troncs de cônes. Cependant, si les troncs de cônes ont des bases perpendiculaires à l'axe, on distingue encore les discontinuités entre les segments de branche consécutifs qui ne sont pas colinéaires (voir Figure 21, image de gauche). On peut alors utiliser des troncs de cône avec une base commune, définie par le plan médian entre les deux segments de branche consécutifs, pour « boucher les trous » (voir Figure 21, image de droite).

Il est à noter que dans ce second cas, il est impossible de définir une seule et même base commune pour une jonction de plus de deux segments de branches (embranchement). Mais on a vu dans le cas des cylindres (Figure 20) que cela ne posait pas de problème particulier puisque la discontinuité devenait alors invisible.

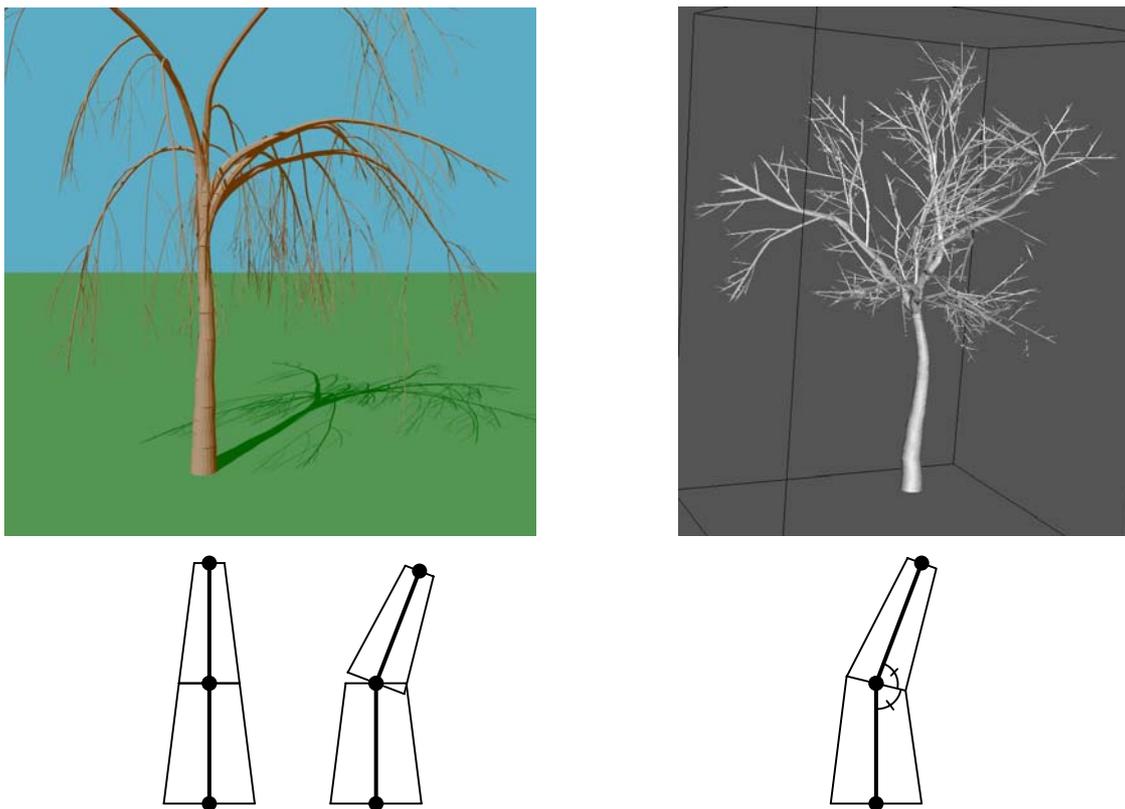


Figure 21 : Arbres formés de troncs de cône. A gauche : chaque tronc de cône a des bases parallèles, ce qui engendre des « trous ». A droite : les troncs de cônes consécutifs ont des bases communes afin de supprimer les discontinuités d'un segment de branche à l'autre.

1.3.3.3 Cône-sphères

La seconde représentation par troncs de cônes corrige tous les problèmes de discontinuité en assurant une continuité C^0 . Néanmoins, outre l'écorce rugueuse, la forme globale des branches dans la nature est plus douce que les angles tranchants visibles à la jonction des segments de branches formés par des segments de cônes. Le cône-sphère introduit par Max [83] permet de corriger en partie ce problème en offrant une continuité C^1 (cela revient géométriquement à ce qu'il n'existe, en chaque point de la surface, qu'un unique plan tangent à la surface, ce qui rend cette dernière plus lisse) à l'extérieur des coudes formés par deux segments de branche successifs. Il propose également une méthode pour adoucir l'intérieur des coudes mais elle est malheureusement peu flexible et n'est pas du tout prévue pour permettre l'application de textures surfaciques 3D telles que l'écorce que nous voudrions y appliquer (voir Chapitre 3). Le principe du cône-sphère est simple : il s'agit de deux sphères (une à chaque extrémité du segment servant de squelette) reliées par un tronc de cône comme le montre la Figure 22 (schéma de gauche).

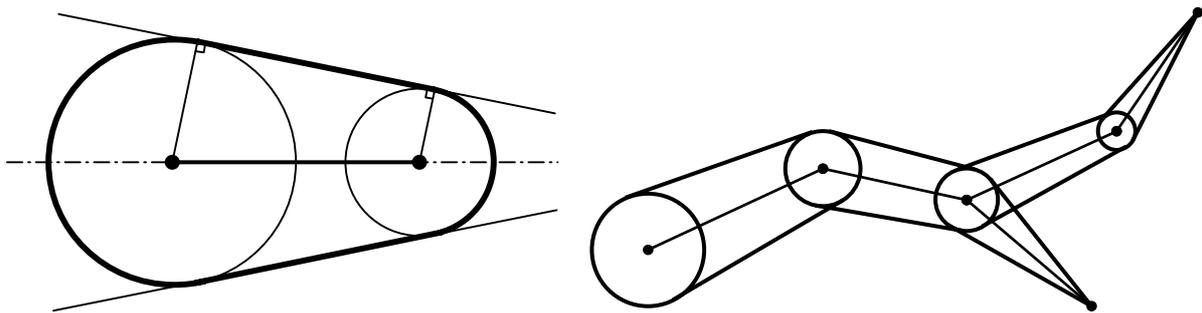


Figure 22 : A gauche : un cône-sphère. A droite : branche formée de cône-sphères.

1.3.3.4 Cylindres généralisés

Une autre manière de représenter des branches sans angles saillants le long de celles-ci est d'utiliser des cylindres généralisés [2] [10] [12]. Tandis qu'un cylindre de révolution classique est défini par l'extrusion d'un cercle de rayon constant le long d'un segment de droite, un cylindre généralisé suit une courbe quelconque formant son squelette (courbe de Bézier [9], « spline » [109], segment de droite, etc.) et sa section n'est pas forcément circulaire et peut varier entre ses deux extrémités.

Un cylindre généralisé, suffisamment simple pour se contenter de corriger les problèmes rencontrés avec les méthodes précédentes, aurait donc une section circulaire dont le rayon décroît de la base du segment de branche à son autre extrémité. Il pourrait alors être représenté par quatre fonctions (x, y, z, r) telles que, pour tout t variant de 0 à 1 entre chaque extrémité de la courbe, $x(t), y(t), z(t)$ donnent les coordonnées du point de la courbe correspondant à t , et $r(t)$ donne le rayon correspondant à ce point.

Cette représentation, peut alors être visualisée de deux manières : de façon globale par surface implicite ou bien en discrétisant le cylindre généralisé en sous-parties plus simples (troncs de cônes ou cône-sphères).

La plus précise de ces représentations est celle par surface implicite ; mais les cylindres généralisés sont cependant assez lourds à afficher de cette façon. Le principe général des surfaces implicites (expliqué plus en détail au paragraphe suivant) nécessite de pouvoir calculer le « potentiel » de n'importe quel point de l'espace par rapport au squelette de l'objet implicite à représenter afin de savoir si ce point est situé à la surface du volume et doit donc être affiché ou bien s'il est à l'extérieur ou à l'intérieur du volume. Ainsi, pour chaque point de l'espace, il faudra déterminer à quel point de la courbe il correspond (généralement on considère que c'est le point le plus proche ; mais, du fait de la possibilité de faire varier le rayon, ce n'est pas forcément le cas, comme le montre la Figure 23 où P dépend de sa projection P_1 et non pas P_2 , bien que $d_2 < d_1$) pour trouver t et ainsi connaître le rayon correspondant et enfin pouvoir calculer le potentiel du point.

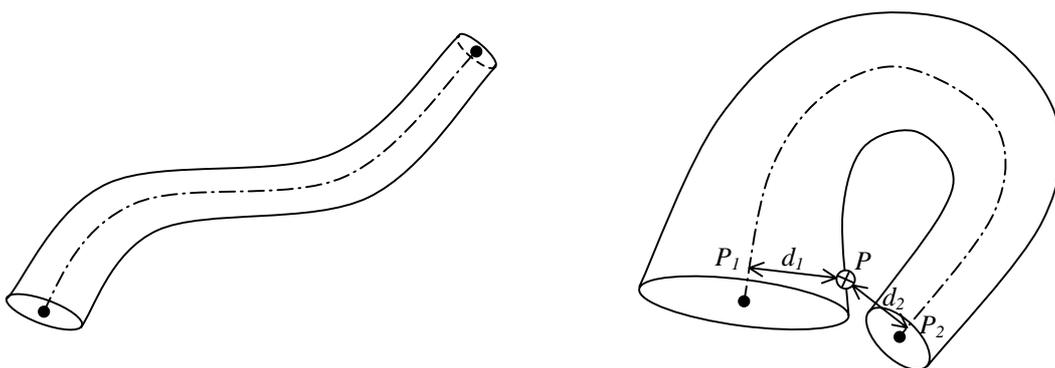


Figure 23 : Cylindres généralisés. Celui de droite illustre un cas particulier où le potentiel d'un point P ne correspond pas au point de la courbe le plus proche.

En pratique, les cylindres généralisés sont donc souvent « discrétisés » en troncs de cônes ou en cônes-sphères en transformant la courbe en une suite de segments et en ne conservant que les rayons qui correspondent aux points échantillonnés sur la courbe (voir Figure 24) ; ceci se faisant avec un pas d'échantillonnage suffisamment faible pour ne pas remarquer le problème d'angles que l'on tentait initialement d'éviter.

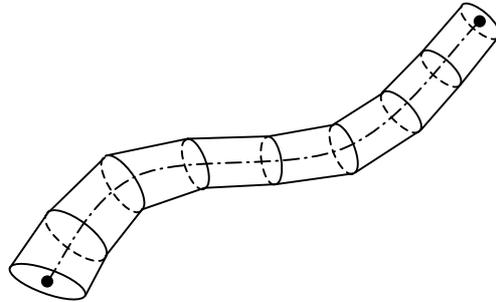


Figure 24 : Cylindre généralisé discrétisé.

Les cylindres généralisés offriraient probablement la meilleure solution s'ils ne présentaient pas, sous forme implicite, une telle complexité de calcul... d'autant plus accrue lorsqu'on souhaite également adoucir les embranchements (et pas seulement les jonctions de segments de branches) tout en y appliquant une texture 3D d'écorce.

1.3.3.5 Surfaces implicites

Nous venons de voir différents moyens pour obtenir des segments de branches (éléments minimaux de la structure de l'arbre) qui s'enchaînent (jonction de deux segments de branche) de façon douce. Néanmoins, cela ne permet pas d'adoucir les embranchements formés d'au moins trois segments de branche, afin de mieux représenter la forme naturelle des embranchements (voir Figure 25). Pour ce faire, il faut recourir au mélange de potentiels (« *blending* ») que seules les surfaces implicites permettent.

Le principe

Le principe de base des surfaces implicites est très simple : à partir d'un squelette S et d'une fonction de distance d (donnant un « potentiel »), chaque ensemble de points ayant le même potentiel forme une « isosurface ». On choisit alors la distance k qui représente

l'isosurface voulue. Ainsi, l'ensemble de tous les points P tels que $f(P,S) = d(P,S) - k = 0$ représente notre surface implicite.



Figure 25 : Photographies d'embranchements sur de vrais arbres. A gauche : une photo d'embranchement et une image illustrant sa silhouette où l'on voit clairement la transition adoucie entre les branches. A droite : un autre exemple d'embranchement où les larges arcs blancs montrent les zones nécessitant d'être adoucies par rapport aux lignes oranges qui montrent les contours qu'aurait l'embranchement sans adoucissement.

Cependant, si chaque segment de branche est défini implicitement et qu'on souhaite adoucir la jonction entre les segments adjacents, alors la simple formule définissant l'isosurface par l'ensemble $\{P \mid f(P,S) = 0\}$ ne suffit plus. Il faut trouver une formule qui mélange (additionne) les potentiels $f(P,S)$ par rapport à chacun des segments implicites S à mélanger.

Or, ce mélange n'est pas évident car il génère un gonflement de la surface à la jonction des volumes implicites puisque la somme des potentiels y croît plus vite. De nombreux travaux sur lesquels nous revenons au paragraphe 3.2 ont vu le jour pour tenter de supprimer ce gonflement. Bloomenthal [13] [14] [15] [16] et Sherstyuk [110] [111] préconisent l'utilisation de « surfaces de convolution », ce qui permet en effet d'atténuer considérablement le gonflement, sans toutefois le supprimer complètement, notamment dans le cas des embranchements où il faut mélanger plus de deux segments de branches. Ferley et al. [37] proposent une autre méthode qui fonctionne mais qui rend l'application de textures difficile. En tout cas, toutes ces méthodes sont malheureusement prévues pour des surfaces lisses.

Le rendu

Pour visualiser des surfaces implicites, les deux principales techniques sont : le lancer de rayons et la polygonisation. Cette dernière se fait en général avec une méthode dite des « *marching cubes* » [78] qui discrétise le volume dans des petits cubes « *voxels* » (généralement stockés dans un arbre octal pour limiter l'utilisation de la mémoire) et permet un rendu assez rapide ; mais elle présente l'inconvénient d'être inadaptée aux surfaces rugueuses comme l'écorce car on devrait pousser trop loin la précision de la polygonisation, ce qui générerait un maillage trop lourd pour représenter tous les détails de l'écorce. Stolte et Caubet [119] [120] ont étudié une façon d'utiliser un rendu par lancer de rayon avec cette représentation discrète de l'objet. Mais, cette structure est encore une fois trop lourde pour un arbre complet avec une texture d'écorce puisque, dans les deux cas décrits ci-dessus (« *marching cubes* » et lancer de rayons discret), il faudrait stocker la totalité de l'arbre en permanence car on pourrait difficilement se permettre de re-discrétiser des parties de l'arbre pour chaque rayon lancé.

Quant au classique lancer de rayons, des méthodes comme celle de Sherstyuk [110] ou de Hart [48], ont été développées afin d'optimiser sa vitesse ; mais elles sont malheureusement destinées aux surfaces lisses. Dans le cas de surfaces quelconques la technique habituelle est le « *ray marching* » [92] qui évalue, pas à pas le long du rayon lancé, le potentiel du point où l'on est pour savoir si l'on vient de traverser la surface implicite. Dès qu'on trouve deux points tels que l'un est à l'extérieur et l'autre à l'intérieur du volume (potentiels de signe différent si l'on a choisi $f(P,S)=0$ comme isosurface autour de S), on recherche par dichotomie la position exacte du point d'intersection du rayon avec la surface. La Figure 26 illustre ce processus.

Dans cet exemple, la recherche pas à pas le long du rayon s'arrête à la 5^{ème} étape et s'ensuit alors une recherche dichotomique entre les points 4 et 5. On peut néanmoins optimiser la première étape de l'approche en commençant à l'intersection du rayon avec un volume englobant simple connu.

Cette méthode peut toutefois oublier des petits morceaux de la surface qui se situeraient entre deux points testés successivement qui seraient tous deux du même côté de la surface. Kalra et Barr [64] puis Hart [49], Hart et Baker [50] et Worley et Hart [135] ont

proposé des variantes assez semblables, qui fonctionnent de plus avec des surfaces implicites rugueuses, pour y remédier tout en ayant une approche optimisée du volume. La méthode présentée par Hart consiste, à chaque étape P_i , à trouver la distance $d(P_i, A)$ entre P_i et la surface implicite A . On en déduit que l'intersection avec le rayon ne peut pas se situer avant d'avoir avancé d'au moins $d(P_i, A)$ le long du rayon, ce qui donne donc la position de la prochaine étape P_{i+1} . Le processus s'arrête donc quand $d(P_i, A) = 0$ ou bien si le rayon ressort du volume englobant.

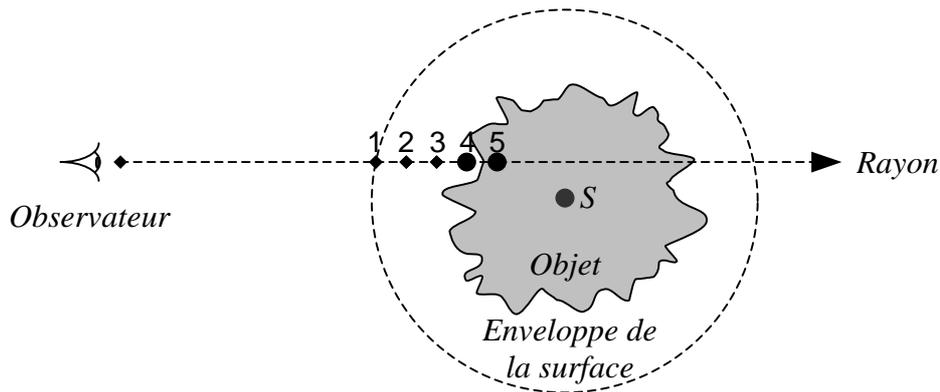


Figure 26 : Principe du « ray marching » sur une surface implicite quelconque.

1.4 L'écorce

Bien que secondaire lorsqu'on observe un arbre de loin et pouvant alors se réduire à une simple couleur (grisâtre, verdâtre, marron, etc.), l'écorce est cependant primordiale pour le réalisme visuel lorsqu'on veut visualiser un arbre de près. Nous allons donc énumérer les différentes méthodes qui ont été élaborées jusqu'à aujourd'hui pour rajouter une texture d'écorce à un arbre.

1.4.1 Placage 2D

La méthode la plus fréquemment rencontrée – et ceci principalement dans les applications en temps réel – est le simple placage de textures 2D. Cette méthode est majoritairement utilisée pour les représentations des branches à l'aide de cylindres ou de troncs de cônes, définis sous forme de polygones ou non. D'une manière générale, cela

signifie qu'à chaque point de la surface des branches correspondra un couple de coordonnées (u,v) pointant sur une couleur dans une texture.

1.4.1.1 Application de texture 2D sur des formes cylindriques

Ces coordonnées (u,v) sont souvent basées sur un repère cylindrique de sorte que la coordonnée u soit cyclique autour de l'axe du segment de branche (et dépendante du nombre de répétitions de la texture autour de la branche) et que la coordonnée v dépende de la distance par rapport à la base de la branche (modulo la hauteur de la texture).

Alors, il faudra néanmoins avoir traité l'échantillon de texture (photographie, dessin ou texture synthétisée) afin que la texture d'écorce puisse être répétée le long des branches sans que l'on remarque de discontinuités sur les bords de la texture ; ce problème se rencontrant principalement dans le cas où la texture serait une photographie. Ce traitement consiste donc à synthétiser, à partir d'un échantillon de texture d'écorce, une texture qui « boucle » à l'aide d'une des multiples méthodes existantes de synthèse de textures [4] [7] [17] [35] [36] [70] [95] [133] [136] [139], comme le montre l'exemple de la Figure 27. Nous verrons plus en détail certaines de ces méthodes au paragraphe 2.3.

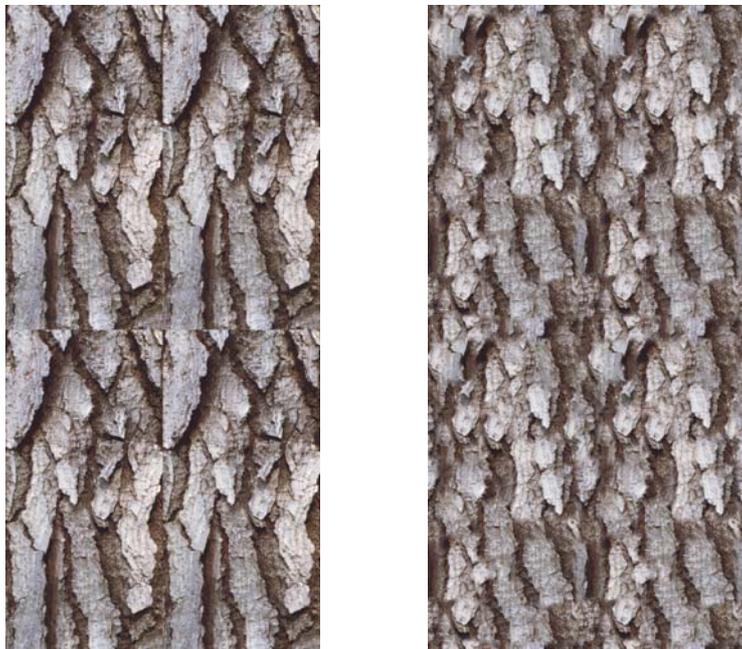


Figure 27 : Juxtaposition d'une texture 2D (appliquée quatre fois) qui ne « boucle » pas (à gauche), puis qui « boucle » (à droite). Dans le premier cas, on constate des discontinuités.

1.4.1.2 Application de texture sur des surfaces quelconques

D'autres méthodes ont été proposées pour appliquer une texture sur une surface quelconque, soit en appliquant des morceaux de texture préalablement choisis par l'utilisateur (« *lapped textures* » [96]), soit en synthétisant la texture directement sur la surface [124] [125] [138] à partir d'un échantillon.

1.4.2 Perturbation de normale (« *Bump mapping* »)

Le placage de texture 2D pose cependant le problème évident d'absence de relief dans la texture. Or, l'écorce est rugueuse dans l'immense majorité des cas. Certes, l'utilisation d'une photographie donne une impression de relief mais elle est trahie par l'éclairage de la scène qui suit de manière régulière la surface lisse des formes cylindriques utilisées pour les branches comme le montre la Figure 28.



Figure 28 : branches recouvertes d'une texture 2D d'écorce dont l'aspect lisse est peu réaliste.

Afin de rajouter une impression de relief, Bloomenthal [12] a modifié la normale en chaque point de la surface des branches (« *bump mapping* » [11]) en utilisant une carte de hauteurs extraite d'un moulage en plâtre d'écorce au moyen d'un scanner à rayons X. La Figure 29 illustre un de ses résultats. Ceux-ci peuvent encore être améliorés en prenant en compte les ombres auto-portées [82].



Figure 29 : Le « Puissant Erable » (« *Mighty Maple* ») de Jules Bloomenthal [12] dont la texture d'écorce est obtenue par perturbation de normale.

1.4.3 Carte de déplacement de surface (« *Displacement mapping* »)

Le déplacement de surface a été introduit par Cook [25] et permet de déplacer chaque point d'une surface suivant une direction donnée à l'aide d'une carte de hauteurs [88]. De nos jours, les cartes graphiques actuelles commencent à le gérer en temps réel et cette méthode remplacera donc probablement à terme la simple perturbation de normale (« *bump mapping* »). En effet, cette technique permet de corriger un important problème qui n'est pas géré par le « *bump mapping* » et qui fait que la silhouette des branches reste parfaitement lisse malgré une impression de relief sur le corps de la branche. Wang et al. [130] ont utilisé cette technique pour visualiser quelques branches modélisées manuellement mais ne proposent pas de méthode pour l'étendre à un arbre complet en raison de la difficulté de paramétrisation de la texture tout au long des branches.

1.4.4 Surfaces implicites perturbées

Du fait d'un niveau de réalisme assez faible offert par la plupart des méthodes utilisant des volumes simples, on s'est assez peu soucié du raccord des textures au niveau des embranchements. Or, c'est un problème qui reste difficile à gérer, la nature elle-même

ne sachant pas le faire de façon douce comme on peut le voir sur la Figure 25. En effet, une partie de la texture doit continuer à suivre la direction de la branche principale tandis que l'autre doit suivre la nouvelle branche.

Bloomenthal [12] n'a fait que constater ce problème sans y apporter de solution. Et nous devons à Hart [50] [51] la plupart des travaux intéressants effectués dans ce domaine, bien qu'il n'offre de solution que pour un double embranchement alors qu'il arrive assez souvent que plus de deux branches partent d'un même embranchement. C'est notamment le cas des arbres dont le tronc s'arrête là où le houppier commence.

L'adoucissement des embranchements et des jonctions de branches ne pouvant se faire qu'avec des surfaces implicites, il est donc intéressant d'arriver à y appliquer une texture et nous avons vu que les textures en relief permettent d'augmenter le réalisme visuel. Ainsi, la première méthode de Hart [50] consiste à représenter les branches de façon implicite et à interpoler la texture d'une branche avec celle de la branche qui la supporte. Les résultats qu'il obtient ne sont malheureusement pas assez réalistes, à cause notamment du fait que la texture d'écorce est synthétisée à l'aide d'une simple fonction de bruit [91]. Chover et al. [23] ont également eu recours au même genre de technique, mais en s'attachant plus à la macro-texture qu'à la micro-texture de l'écorce. La Figure 30 montre un exemple de résultat obtenu par Hart. D'autre part, le problème de gonflement mentionné au paragraphe 1.3.3.5 n'est pas pris en compte car il n'intervient pas lorsqu'on mélange une petite branche avec une grosse si le champ de potentiel de la petite branche est plus étroit que la grosse branche, ce qui est le cas dans les exemples de Hart.



Figure 30 : Jonction par mélange de deux branches implicites (Hart [50]).

Sa seconde méthode [51], en association avec Baker, est basée à la fois sur des surfaces implicites mais également sur des éléments géométriques disposés à la surface, ce qui nous amène au paragraphe suivant : les « *texels* ».

1.4.5 Texels

Les « *texels* » sont des éléments de texture que l'on peut disposer sur une surface. Dans le cas qui nous intéresse, ces éléments représenteraient donc des petits morceaux d'écorce. Ainsi, la technique de Hart et Baker [51] consiste à faire parcourir des particules le long des branches, avec une gestion spéciale de séparation et de regroupement aux embranchements, et place des petits *texels* le long du parcours de chaque particule. Les résultats visuels sont bons, même si le réalisme n'est toujours pas parfait (voir Figure 31). Mais surtout, cette méthode ne s'adapte qu'à un nombre très restreint des types d'écorces que l'on rencontre dans la nature (cf. section 1.2.3) et il est difficile de modéliser manuellement les *texels* nécessaires à la représentation d'un type d'écorce particulier.

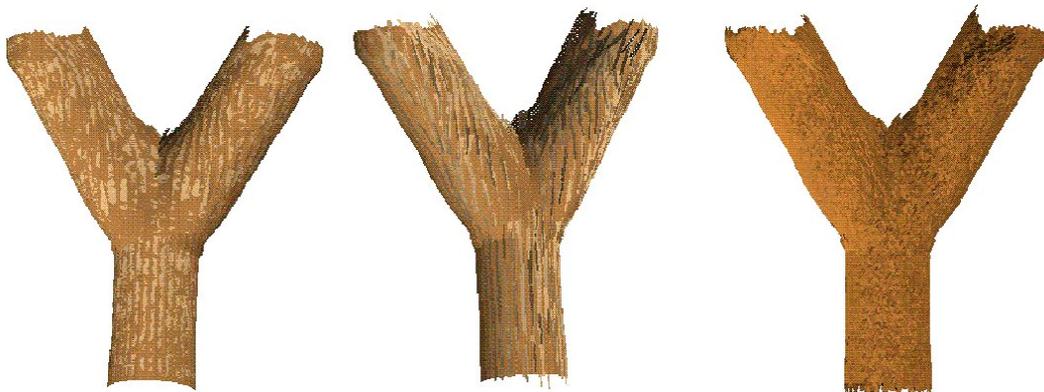


Figure 31 : Résultats obtenus par Hart et Baker en recouvrant des branches avec des *texels*.

Fleischer et al. [39] ont également proposé une méthode pour recouvrir de *texels* (cellules) une surface arbitraire – et donc adaptable aux arbres – même si, une fois encore, le problème de la modélisation de ces cellules se pose.

1.4.6 Synthèse des craquelures par système de masses-ressorts.

Plus récemment, Lefebvre et Neyret [67] se sont penchés sur le développement de l'écorce le long des branches à l'aide d'un système de masses-ressorts. Ils donnent au

système quelques paramètres définissant les caractéristiques de l'écorce et les déchirures se forment d'elles-mêmes. Le réalisme visuel est assez bon car il est garanti par l'utilisation de photos d'écorce pour représenter les deux niveaux principaux de l'écorce : la surface et le fond des craquelures. De plus, le problème de la continuité de la texture aux embranchements a été pris en compte et fonctionne plutôt bien. La Figure 32 montre différents résultats obtenus par Lefebvre et Neyret.



Figure 32 : Divers résultats obtenus avec la méthode de synthèse de déchirures de Lefebvre et Neyret.

Toutefois, bien qu'il soit également possible de rajouter des niveaux de craquelures supplémentaires pour rajouter plus de détails, ce point reste perfectible car, outre les changements de hauteur entre la surface de l'écorce et le fond des déchirures, le reste de l'écorce reste plat. De plus, la méthode utilisant des cylindres généralisés ne permet pas a priori d'adoucir les intersections de branches. Enfin, la méthode générant un nombre assez important de polygones, la représentation d'un arbre complet est très lourde.

1.5 Les feuilles

Lorsqu'on observe un arbre de près, les feuilles comptent beaucoup dans le réalisme du rendu en images de synthèse. Nous verrons ici quelles sont, à ce jour, les différentes façons de modéliser et de visualiser des feuilles en Infographie.

1.5.1 Modélisation (synthèse) des feuilles

Il y a plusieurs années, lorsque les textures en haute résolution étaient utilisées avec parcimonie en raison de leur poids en mémoire, de nombreuses méthodes sont apparues pour tenter de modéliser intégralement différents types de feuilles.

Les feuilles étant relativement plates, leur principale caractéristique est la forme de leur limbe. Une première façon intuitive serait donc de reproduire manuellement le contour d'une feuille sous forme d'un polygone avec un grand nombre de côtés. Mais, si cette méthode peut paraître simple pour certains types de feuilles, elle l'est nettement moins pour les feuilles ayant un contour plus complexe, comme les feuilles lobées, dentées, etc. (cf. paragraphe 1.2.4.1). C'est pourquoi un certain nombre de méthodes ont été étudiées pour modéliser le contour des feuilles de façon automatique.

Toutefois, la modélisation manuelle reste encore possible grâce à des interfaces graphiques proposant des outils pour modéliser rapidement le contour et la courbure des feuilles, notamment grâce au logiciel de Deussen et Lintermann [75] [76] [77].

Parmi les méthodes automatiques, on peut citer Oppenheimer [87] qui a utilisé les propriétés « pseudo-fractales » des nervures, très proches de la structure des branches d'arbres, pour modéliser des feuilles d'une assez grande complexité mais qui ont cependant une structure visiblement très répétitive. Prusinkiewicz et al. [98] se sont servis des L-systèmes pour tracer le contour des polygones pouvant représenter très précisément différents types de feuilles.

Le contour du limbe est souvent lié aux nervures. Ainsi, pour modéliser le limbe, il peut être utile de modéliser d'abord les nervures. C'est notamment sur cette idée que Viennot et al. [127] ont modélisé les nervures principales et secondaires à l'aide de paramètres tels que le nombre de nervures principales (et donc de lobes), les angles entre les nervures, etc. Puis, ils ont déduit le contour de la feuille en fonction d'une distance par rapport aux extrémités des nervures. Janey [60] a repris cette représentation et l'a étendue à l'aide de courbes (fonctions cubiques) pour adoucir les bords du limbe, y ajouter des dentelures, etc. Il y a également rajouté des dégradés de couleurs.

Se basant sur des nervures modélisées à base de L-systèmes, Hammel et al. [47] retrouvent le contour de façon implicite.

Bloomenthal [15] a reconstruit le limbe à partir de nervures primaires et secondaires tracées manuellement. Il a ensuite donné du volume aux nervures sous forme de surfaces implicites mélangées à leurs jonctions, de la même manière que pour des branches d'arbres.

Enfin, Lienhardt [71] ainsi que Lienhardt et Françon [72] utilisent un modèle sous forme de carte planaire modulaire. La feuille est représentée de façon topologique plutôt que géométrique et elle est composée d'arêtes, de sommets et de faces ayant des caractéristiques et un rôle précis. La croissance de la feuille peut alors être simulée en appliquant certaines opérations sur ces éléments. Au final, cette méthode permet de représenter les différentes étapes du développement d'un très grand nombre de feuilles.

1.5.2 Acquisition par photographies

De nos jours où une texture en assez haute résolution peut être stockée sans problème en mémoire, le meilleur moyen de modéliser une feuille est sans doute d'utiliser une vraie feuille numérisée à l'aide d'un scanner à plat ou d'un appareil photo numérique et de la détourer, ce qui peut se faire en quelques secondes à l'aide d'un logiciel de retouche d'images, voire automatiquement. On peut alors plaquer cette texture comportant des zones de transparence sur un ensemble réduit de polygones qui englobe la forme de la feuille (voir Figure 33) afin d'avoir une feuille photo-réaliste (puisque c'est une photo !) et avec un contour parfait, pour peu que la résolution de la numérisation soit suffisante. Cette méthode a été utilisée dès le milieu des années 80 par Bloomenthal [12].

Les seuls inconvénients de cette méthode sont que l'éclairage utilisé lors de l'acquisition ne sera pas forcément le même que lors du rendu, ce qui risque de générer des ombrages faux, et d'autre part que le relief des nervures n'est pas pris en compte. Mais, les détails de la texture qu'offre la photographie apportent malgré tout énormément au réalisme du rendu, ce dont manquent encore les méthodes synthétisant entièrement les feuilles.

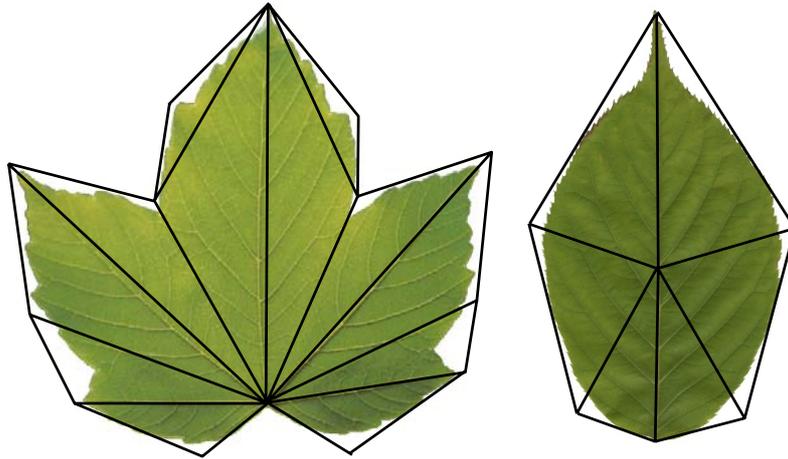


Figure 33 : Feuilles modélisées à partir de textures détournées plaquées sur un ensemble réduit de polygones.

1.5.3 Travaux sur la translucidité et la réflectance

Les principaux travaux, dans le domaine de la transmission de la lumière au travers de tissus végétaux tels que les feuilles des arbres et de la réflexion de la lumière sur ceux-ci en Infographie, ont été réalisés par Baranoski et Rokne [5] [6]. Ils ont présenté un modèle de transmittance et de réflectance pour les feuilles basé sur des notions de Botanique et de Physique et, entre autres, sur les expériences menées par Woolley [134]. La Figure 33 illustre un exemple des résultats obtenus par Baranoski et Rokne.

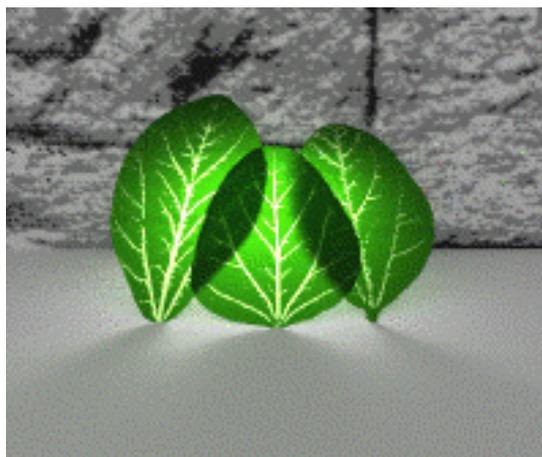


Figure 34 : Simulation de la transmission de la lumière à travers des feuilles.

1.6 Conclusion

Comme nous l'avons vu dans la section 1.2 (Biologie), la diversité des espèces d'arbres, jusqu'à la variété de ses détails, principalement l'écorce mais également les feuilles, rendent leur représentation informatique difficile. De plus, leur complexité les rend lourds à traiter.

Des méthodes impressionnistes ont été proposées dans le passé. Même si certaines d'entre elles sont toujours d'actualité pour les applications temps-réel, aucune de ces solutions n'est prévue pour le rendu réaliste et encore moins de près.

Dans le processus de rendu d'un arbre en images de synthèse, on trouve en premier lieu sa modélisation (cf. paragraphe 1.3.2), pour laquelle les meilleurs résultats sont obtenus en modélisant le squelette des branches auquel on donne ensuite du volume. Sur ce point, les résultats actuels sont déjà excellents et ne nécessitent plus d'améliorations particulières, en ce qui concerne le réalisme visuel de la forme des arbres en tout cas.

La seconde tâche consiste à rajouter des feuilles (cf. section 1.5) au bout de ces branches. Dans ce domaine, on différencie la modélisation et le rendu. Les principaux problèmes portant sur les feuilles ont déjà été plus ou moins résolus. Différentes techniques de modélisation ont été définies mais, la plupart du temps, on a recours à une texture de feuille plaquée sur un ensemble plus ou moins grand de polygones représentant une feuille, le plus souvent modélisée assez rapidement par l'utilisateur. En ce qui concerne le rendu, les meilleurs résultats sont obtenus lorsqu'on utilise une image numérisée d'une vraie feuille, ce qui rend inappropriées les méthodes de modélisation complète des feuilles. De plus, des travaux ont été réalisés sur la transmission de la lumière à travers les feuilles afin d'obtenir un rendu plus fidèle. Les principales améliorations que l'on pourrait encore apporter sur les feuilles portent essentiellement sur les nervures qui pourraient être utilisées pour faire varier le relief et la translucidité des feuilles. En effet, d'une part les nervures principales ressortent de la surface plate des feuilles, mais, d'autre part, elles ont des propriétés optiques différentes du reste du limbe et laissent passer la lumière différemment. C'est sur les nervures que nous nous proposons de nous pencher. Et comme, aujourd'hui, les nervures sont soit modélisées automatiquement, ce qui empêche alors d'y associer une photographie d'une feuille numérisée, soit dessinées à la main, ce qui est fastidieux, nous

nous proposons d'établir une méthode pour retrouver automatiquement le relief des nervures associées à l'image de feuille employée.

Ces deux premières étapes, si l'on exclue les améliorations du rendu des feuilles, sont déjà nécessaires pour synthétiser un arbre vu de loin. Mais si l'on souhaite avoir un rendu réaliste même de près, il faut alors se pencher sur les transitions entre les branches (paragraphe 1.3.3) et sur l'écorce (paragraphe 1.4). Ce seront ces deux points que nous aborderons principalement au cours de cette thèse car ce sont les plus critiques dans le rendu réaliste d'arbres vus de près.

En ce qui concerne les branches, de nombreux travaux ont été réalisés pour représenter des structures arborescentes avec des transitions douces aux embranchements. Cependant, outre quelques problèmes de gonflements aux intersections de branches qui n'ont toujours pas été complètement résolus, presque toutes ces méthodes ont été prévues pour des surfaces totalement lisses alors que nous voudrions pouvoir y ajouter de l'écorce rugueuse afin de maximiser le réalisme visuel. D'autre part, les quelques méthodes permettant de rajouter du relief à ces surfaces ne permettent pas vraiment de contrôler précisément sa profondeur. Nous présenterons donc une méthode permettant de résoudre tous ces problèmes car c'est aujourd'hui ce dernier domaine – l'écorce – qui nécessite le plus d'amélioration dans le rendu réaliste des arbres.

Les améliorations nécessaires à l'obtention d'un meilleur rendu étant plus nombreuses et plus importantes pour l'écorce et les branches que pour les feuilles, nous consacrerons des chapitres plus larges à ces deux premiers thèmes.

Chapitre 2

Analyse et synthèse d'écorce

2.1 Introduction

Nous avons constaté au chapitre précédent que l'écorce est un élément essentiel dans la qualité et le réalisme du rendu d'un arbre vu de près. Pour finir de s'en convaincre, il suffit d'observer la Figure 35 qui montre, d'une part, qu'un simple placage de texture 2D (image du milieu) ne suffit pas à courte distance car l'observateur s'attend à voir une texture rugueuse et, d'autre part, ce que pourraient apporter les textures en relief (image de droite) en termes de réalisme visuel.



Figure 35 : A gauche : arbre virtuel observé de loin pour lequel les détails de l'écorce ne sont pas importants. Au centre : vue rapprochée (et, en médaillon, encore plus rapprochée) sur des branches d'un arbre texturé avec un placage de textures 2D traditionnel. A droite : un résultat significativement amélioré grâce à notre technique utilisant de l'écorce en relief (et une transition adoucie aux intersections de branches que nous présentons au Chapitre 3).

La perturbation de normale (« *bump-mapping* ») [11], contrôlée par une carte de hauteurs permet d'améliorer le rendu de l'écorce, comme l'a fait Bloomenthal [12] (cf. paragraphe 1.4.2), en particulier si les ombres auto-portées [82] sont prises en compte ;

mais cela ne permet pas de traiter la silhouette rugueuse des branches. Par conséquent, une approche plus adéquate serait d'utiliser des techniques d'application de textures 3D telles que le déplacement de surfaces (« *displacement mapping* ») [25] [88] [130] ou encore « l'*hypertexture* » [92] [31]. Le problème à résoudre porte donc sur la modélisation de ces textures 3D d'écorce, soit par acquisition soit par synthèse.

La méthode donnant a priori les meilleurs résultats en vue d'obtenir une texture en relief est évidemment celle faisant appel à un scanner 3D. Ainsi Bloomenthal [12] a utilisé un scanner à rayons X sur un moulage en plâtre d'un tronc d'érable. Cette méthode a cependant l'inconvénient de ne récupérer que le relief et pas les couleurs de l'écorce, qui apportent pourtant une information essentielle au réalisme visuel. Il existe de nos jours des appareils permettant d'acquérir à la fois la couleur et le relief d'un objet ; mais ce ne sont encore que des solutions marginales en raison de leur coût et ne sont donc pas encore accessibles à tous.

On s'orientera alors vers la synthèse de textures d'écorces 3D. Une fonction de bruit [91] peut être utilisée pour perturber une surface implicite. C'est la solution qui a été retenue par Chover et al. [23] et par Hart [50] pour obtenir une texture rugueuse qui ressemble plus ou moins à de l'écorce. D'autre part, Hart et Baker [51] ont décrit une technique pour générer un petit nombre de sortes d'écorces en 3D au moyen de particules qui parcourent les branches en longueur. Cependant, ces différentes approches sont limitées à un nombre réduit d'espèces virtuelles (imaginaires) d'écorce et manquent donc de réalisme en comparaison à des méthodes qui seraient basées sur des modèles d'écorces réelles, comme par exemple des photographies ou des mesures.

En fait, il semble difficile de synthétiser des textures 3D comme l'écorce, qui soient suffisamment réalistes visuellement, sans avoir recours à des méthodes basées sur des modèles de textures réelles. C'est à partir de ce constat que Dischler et al. ont, par exemple, synthétisé diverses textures 3D en relief [30] ou solides [29] en analysant des photographies. Et, puisque cette observation n'est donc pas seulement vraie pour l'écorce mais aussi pour beaucoup de textures naturelles (et d'autres encore), des méthodes comme celles de Dana et al. [26] et Rushmeier et al. [108] proposent d'acquérir des textures directement à partir de l'analyse de séries de photographies d'un même objet réel texturé, prises sous différentes conditions de vue et d'éclairage. Malheureusement ces solutions

sont compliquées et peu pratiques à mettre en œuvre, surtout avec de l'écorce réelle, car elles nécessitent de pouvoir placer l'objet au centre de l'installation, ce qui n'est pas facile avec l'écorce pour laquelle il faudrait, soit monter le matériel en extérieur, soit découper un tronc d'arbre. Par conséquent, des méthodes basées sur une seule photographie semblent plus viables et permettent de surcroît de bénéficier des importantes bases de données que représentent les collections de textures sur CD-ROM et sur Internet, ainsi que les livres tels que celui de Vaucher [126]. Par conséquent ceci rend possible la représentation d'arbres difficiles à trouver physiquement dans la nature, comme des arbres exotiques ou rares.

Une autre approche, un peu à part mais intéressante, est celle de Lefebvre et Neyret [67], présentée plus en détail au paragraphe 1.4.6, qui consiste à synthétiser les craquelures à partir d'un modèle physique de masses-ressorts et à augmenter le réalisme en appliquant des textures 2D différentes, provenant de photographies d'écorce, aux différents niveaux définis par les craquelures. Nous avons cependant préféré baser tous nos travaux de ce chapitre sur l'extraction du relief à partir de photographies car nous pensons que c'est l'ensemble des détails de l'écorce qui caractérisent son aspect et pas seulement ses craquelures qui ne forment en quelque sorte que sa macro-texture et qui ne sont présentes sous la forme décrite par Lefebvre et Neyret que sur certaines espèces d'arbres.

Même si l'extraction de la couleur d'une texture d'écorce à partir d'une photographie n'est pas forcément triviale – si l'on souhaite en retirer les ombrages dus au relief, pour ne conserver que la vraie couleur – le problème principal dans le cadre des textures 3D reste tout de même l'extraction du relief. Ce chapitre commence donc (au paragraphe 2.2) par une description des méthodes que nous avons envisagées et, pour certaines, retenues pour extraire l'information de relief (et de couleur) à partir d'une simple photographie d'écorce.

Cependant, il n'est pas suffisant de retrouver le relief correspondant à une photographie pour pouvoir appliquer cette texture 3D sur les branches d'un arbre (voir Chapitre 3). En effet, la simple juxtaposition de l'échantillon de texture 3D le long et autour des branches génère à la fois des discontinuités visibles et gênantes mais aussi une impression de répétition du même motif qui ne paraît pas naturelle. Il faut donc pouvoir synthétiser une texture plus large, voire infinie, qui soit applicable aux branches ; ce qui fait l'objet du paragraphe 2.3.

Enfin, nous apportons nos conclusions sur le thème de l'analyse et de la synthèse de textures d'écorces par rapport aux méthodes que nous avons retenues, en parlant de leurs avantages, de leurs limites et des éventuelles améliorations à y apporter.

2.2 Extraction du relief à partir d'une photographie

Toutes les méthodes décrites ici partent de l'observation qu'il y a une forte corrélation entre la luminosité (les niveaux de gris) d'une photographie d'écorce et le relief de cette écorce. Cette idée est développée au paragraphe 2.2.1.

La première méthode qui vient à l'esprit lorsqu'on parle de récupérer le relief en analysant la luminosité d'une image est le « *shape-from-shading* » (abrégé en « SFS » par la suite), dont Horn et Brooks [57] puis Zhang et al. ont fait d'intéressants tours d'horizon. Le SFS dont nous discutons au paragraphe 2.2.2 peut être traduit littéralement par « forme à partir des ombrages ». La plupart des algorithmes de SFS ont cependant des contraintes fortes quant aux propriétés de la surface à analyser. La texture doit le plus souvent être lisse et monochrome, ce qui n'est pas le cas de la plupart des écorces.

Nous présentons donc ensuite, au paragraphe 2.2.3, une première méthode empirique que nous avons proposée [81] puis améliorée et qui consiste à prendre avantage de la rugosité des écorces en appliquant un certain nombre de traitements d'image simples sur le modèle en niveaux de gris pour obtenir, in fine, une image qui puisse être utilisée comme carte de relief du modèle fourni. Cette technique fonctionne assez bien pour une majorité des espèces d'écorces mais pas pour toutes. Nous avons donc cherché d'autres méthodes pour gérer les catégories d'écorce restantes. Le paragraphe 2.2.4 traite d'une autre méthode que nous avons proposée [32] dont le but est toujours d'extraire une carte de hauteurs cohérente par rapport au modèle ; mais, cette fois-ci, en différenciant les deux niveaux de détail que sont la macro-texture, représentant la structure principale des motifs de l'écorce, et la micro-texture qui contient les détails. Enfin, une méthode reprenant les idées des deux précédentes est présentée au paragraphe 2.2.5. Notons que, la première et la dernière de ces trois méthodes utilisent chacune leur propre classification, regroupant la plupart des écorces existantes, en se basant sur des observations visuelles et botaniques telles que celles du paragraphe 1.2.3.2, afin d'apporter des données communes à chaque catégorie

fournissant ainsi les données manquantes qu'il est difficile ou impossible d'extraire d'une photographie. L'utilisateur n'aura donc qu'à préciser la catégorie à laquelle appartient l'écorce dont il fournit une photographie.

Il est cependant important de noter que, pour toutes ces méthodes, notre but n'a pas été de retrouver le relief exact du modèle mais d'obtenir une carte de hauteurs qui soit cohérente avec le modèle 2D fourni ; c'est-à-dire que cette carte puisse donner l'illusion qu'elle correspond bien au modèle. Les résultats obtenus sont néanmoins visuellement très réalistes pour la plupart des types d'écorces.

2.2.1 Corrélation entre les niveaux de gris et le relief

Parmi les composantes formant la couleur de chaque pixel d'une image, la luminosité est celle qui représente le mieux la plupart des images. Elle est représentée par les niveaux de gris qui peuvent être considérés de deux façons :

- La première est de se dire que les zones claires de la texture sont celles qui sont face à la lumière tandis que les zones plus sombres sont celles que la lumière n'atteint pas ou peu. Ainsi, si l'on connaît la direction de la source lumineuse et les propriétés de réflexion (« albédo ») de la texture, on peut retrouver, par rapport à cette source de lumière, l'orientation de chaque quadrilatère correspondant à chaque pixel de l'image et formant la surface. Il est d'ailleurs parfois possible de déterminer automatiquement la direction de la lumière [89] [141] [142] en analysant l'image. On peut ensuite tenter de recomposer le relief en ajustant au mieux les hauteurs de ces pixels pour que les quadrilatères associés se joignent au mieux. C'est l'idée de base de la plupart des algorithmes du « *shape-from-shading* » (SFS) [57] qui est traité au paragraphe suivant.
- La seconde approche est de constater que, bien souvent, les zones les plus sombres sur une photographie d'écorce représentent plus ou moins les zones les plus profondes. Cette observation est cependant approximative et comporte de nombreuses exceptions comme par exemple l'écorce de bouleau où les zones sombres représentent des protubérances qui ressortent par rapport aux zones blanches plus lisses. De plus, dans le cas général, la façon dont est éclairée l'écorce joue a priori un rôle important car, suivant la direction de la lumière, les zones sombres ne seront pas au même endroit.

Néanmoins, pour cette deuxième approche, la structure principale de l'écorce ne se trouvera pas radicalement changée par l'éclairage. En effet, les zones d'ombre sont toujours moins élevées que les sommets de l'épiderme. Et, comme nous allons le montrer par la suite et notamment au paragraphe 2.2.3, cela permet – avec quelques réglages – d'obtenir des résultats qui restent visuellement cohérents avec le véritable relief de l'écorce, même s'il est extrêmement difficile d'extraire précisément ce dernier à partir d'une simple photographie (par définition en deux dimensions). Cependant, le but recherché étant avant tout le « photo-réalisme » et non le « réalisme absolu » (voir discussion au chapitre d'introduction de la thèse), nous nous satisfaisons de ces résultats.

Dans un premier temps, nous avons tenté de justifier notre hypothèse disant que, dans le cas simple d'une texture monochromatique (ce qui exclue les cas particuliers comme le bouleau), les zones les plus profondes sont globalement les plus sombres car elles reçoivent a priori moins de lumière. Si cette hypothèse est vraie, alors l'illumination globale d'une surface équivalant à la texture en niveaux de gris, où les pixels sombres auraient une profondeur inférieure aux pixels clairs, devrait être visuellement très proche de la texture elle-même, prouvant ainsi l'équation *profondeur ≈ obscurité*.

Nous avons donc pour cela pris quelques images en niveau de gris de textures d'écorces monochromatiques avec des craquelures assez prononcées et se prêtant donc a priori assez bien à l'expérience. Pour chacune de ces textures, l'image a servi de carte de hauteurs pour représenter une texture dont nous connaissions donc parfaitement le relief. Ce relief a été illuminé de deux manières différentes pour voir si le résultat obtenu ressemble à la carte de hauteurs sans être influencée par le type d'éclairage. Nous avons tout d'abord simulé une illumination globale avec une lumière diffuse car c'est le cas général en forêt (ou par temps nuageux) où la lumière du soleil atteint rarement directement le tronc des arbres et parce que cela permet d'éviter au maximum les ombres nettes. Nous avons ensuite procédé à un éclairage directionnel pour simuler la lumière directe du soleil dans le cas d'un arbre isolé et pour obtenir cette fois-ci des ombres portées assombrissant en partie le fond des craquelures.

Les résultats obtenus (voir Figure 36) ont montré que l'hypothèse de base (*profondeur ≈ obscurité*) fonctionne plutôt bien (même si nous verrons au paragraphe

suivant comment améliorer ces résultats) pour un éclairage directionnel, que ce soit dans le cas où les craquelures sont profondes et étroites ou non (dans une certaine limite). Toutefois, pour un éclairage diffus, nous avons pu observer que le fond des craquelures est plus éclairé que les bords, ce qui peut donner l'impression qu'il y a deux fois plus de surface en relief que sur le modèle. Ce phénomène s'accroît lorsque les craquelures sont un peu plus larges. C'est compréhensible car, si les bords des craquelures sont éloignés, ils cachent peu la lumière ; et le fond est alors presque autant éclairé que les sommets de l'épiderme, tandis que les bords plus verticaux des craquelures réfléchissent moins la lumière, puisque moins de lumière leur arrive de face. Enfin, tous les résultats présentent l'inconvénient d'être plus rugueux/chaotiques que le modèle.

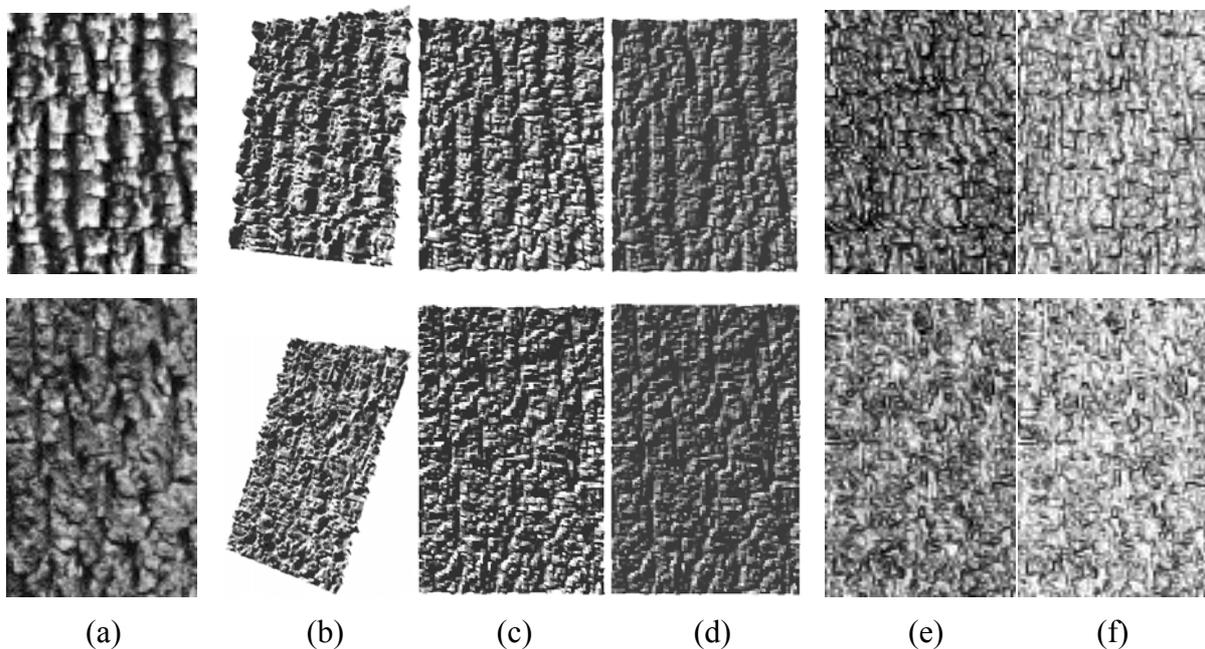


Figure 36 : Illumination d'une carte de hauteurs (connue) représentant une texture d'écorce (une par ligne) afin de comparer les ressemblances entre la carte de hauteur et le résultat illuminé. (a) Modèle connu (photographie d'écorce) utilisé comme carte de hauteurs. (b) Illumination directionnelle avec une source lumineuse venant du coin haut-gauche. (c) Même illumination que (b) mais vue de face. (d) Même vue que (c) mais avec une élévation plus basse de la texture pour simuler un élargissement (relatif) des craquelures. (e) Illumination globale diffuse avec la même élévation que sur les images (b) et (c). (f) Même vue que (e) mais avec la même élévation que (d).

Mais alors pourquoi a-t-on tout de même l'impression, en observant diverses textures d'écorces (peu importe la façon dont elles sont illuminées), comme par exemple les deux modèles de la Figure 36, que le fond est plus sombre ? Il y a en fait deux raisons principales à cela :

- Dans la nature, l'éclairage est rarement parfaitement diffus. Il y a toujours une (ou parfois plusieurs) direction(s) plus ou moins privilégiée(s) d'où vient la lumière. Et, à moins qu'elle vienne du zénith par rapport à l'écorce, les bords des craquelures auront toujours tendance à faire un peu d'ombre.
- Mais surtout, la poussière et les diverses particules de saleté qui flottent dans l'air se déposent sur l'écorce et restent au fond des craquelures et noircissent avec le temps.

On peut donc en effet considérer que dans un grand nombre de cas la profondeur du relief peut être approximativement exprimée par l'obscurité des niveaux de gris de l'image servant de modèle. Toutefois, il n'est pas possible de généraliser ce constat à toutes les écorces mais uniquement à celles ayant des craquelures ou qui soient suffisamment chaotiques comme le sont respectivement (première ligne puis deuxième ligne) les deux exemples de la Figure 36.

Plus généralement, même si la luminosité permet de différencier les niveaux de hauteur, on ne peut pas forcément en déduire le relief par la règle précédente comme le montre l'exemple du bouleau. Nous présentons donc, dans les paragraphes suivants, des méthodes qui peuvent être employées pour traiter un maximum de cas avec un minimum d'effort pour l'utilisateur et qui permettent de retrouver une carte de hauteurs correspondant visuellement à la texture d'écorce fournie en se basant sur les niveaux de gris du modèle, en les interprétant, soit comme l'élévation, soit comme indicateurs de la structure de la texture permettant de retrouver les éléments formant le relief.

2.2.2 Retrouver le relief à partir de l'illumination : « *Shape from Shading* »

Les algorithmes de "shape-from-shading" (SFS) [57] [140], tels que ceux de Funt [40], Leclerc et Bobick [66], Lee et Kuo [68], Lee et Rosenfeld [69], Pentland [90] ou encore Tsai et Shah [123], analysent les niveaux de gris d'une image en supposant qu'ils représentent une information d'ombrage. Un pixel clair signifie que la normale à la surface

est proche de la direction de la source lumineuse et un pixel sombre signifie soit que la normale n'est pas dirigée vers la lumière, soit que le pixel est assombri par l'ombre d'un bout d'écorce. Cette ambiguïté, due aux ombres auto-portées, est un problème et c'est une des raisons pour lesquelles les algorithmes de SFS imposent la contrainte d'avoir une surface suffisamment lisse pour éliminer la présence de ces ombres. Cet aspect lisse est également nécessaire pour ne pas trop perturber les méthodes dites « globales » de SFS, chargées d'équilibrer les niveaux de chaque pixel en fonction de l'orientation de chaque micro-facette représentée par un pixel. Donc, plus les pixels voisins sont différents, plus il sera difficile de retrouver la surface sous-jacente.

Un autre problème est que le SFS ne fonctionne généralement qu'avec des images monochromes, même si certaines images colorées peuvent être traitées si les changements de chromaticité sont abrupts [40]. Enfin, la plupart des méthodes ne fonctionnent que sur des surfaces Lambertiennes (parfaitement diffuses) ou du moins des surfaces à albédo constant et connu, mais ce n'est pas problématique pour les écorces car elles sont le plus souvent assez diffuses.

Malheureusement, l'écorce n'est généralement pas assez lisse pour supposer qu'il n'y aura pas d'ombres portées et peu d'écorces ont des changements de teintes brusques. De plus, Zhang et al. [140] ont étudié un grand nombre d'algorithmes de SFS et en ont conclu que même dans les conditions idéales, très éloignées des écorces, il est difficile d'obtenir des résultats satisfaisants.

2.2.3 Extraction du relief par traitements sur les niveaux de gris de l'image

Compte tenu des contraintes importantes du SFS (cf. paragraphe précédent), souvent incompatible avec la rugosité de la plupart des écorces, nous avons décidé de tirer profit de cette rugosité qui nous a permis de dresser le constat du paragraphe 2.2.1, indiquant que les niveaux de gris du modèle photographique d'écorce permettent très souvent de déterminer une approximation exploitable du niveau de profondeur de la surface sous-jacente.

Notre but est d'analyser une simple photographie de n'importe quelle écorce donnée par l'utilisateur, à partir du plus grand nombre possible d'espèces d'arbres, pour déterminer son relief et ensuite synthétiser une texture 3D d'écorce visuellement similaire en vue de

l'appliquer sur des branches d'arbre. Du fait que nous utilisons une photographie unique comme modèle, nous ne pouvons calculer qu'un relief en 2D½ représenté par une carte de hauteurs qui assigne une élévation à chaque pixel du modèle 2D. Par conséquent, quelques types d'écorces présentées dans la Figure 9 du paragraphe 1.2.3.2 – en particulier parmi celles dont des morceaux se détachent par exfoliation – ne sont pas supportées par la méthode que nous proposons ici.

Si l'on considère la très large variété des écorces, il est très difficile de proposer une méthode générale qui fonctionne pour toutes les écorces. Cependant, si l'on connaît le « type » de l'écorce à traiter, on peut alors évaluer la plupart des données qui ne peuvent pas être extraites à partir de la photographie fournie. Par « types d'écorces », nous n'entendons pas « espèces d'écorces », dont le nombre est du même ordre que celui des espèces d'arbres, mais « catégories d'écorces » regroupant des espèces d'écorces qui peuvent être visuellement assez différentes mais qui pourront être traitées de la même manière. Par exemple : « écorces à protubérances sombres » peut être une catégorie qui regroupera diverses écorces dont les zones les plus sombres ressortent. Nous avons donc re-classifié les écorces en un nombre le plus petit possible de groupes ayant des propriétés « similaires ». Cette classification, présentée au paragraphe 2.2.3.3, est assez semblable à celle à bases botaniques de la Figure 9. Ainsi, l'utilisateur n'aura qu'à fournir une photographie numérisée d'écorce et indiquer son type dans notre classification. L'extraction du relief de cette photographie se fera alors automatiquement grâce à la méthode détaillée dans les sous-paragraphe suivants en tenant compte des connaissances additionnelles apportées par le type sélectionné.

Ci-après, nous commençons par expliquer les conditions dans lesquelles la photographie à fournir doit être acquise. Ensuite, nous décrivons l'idée de base de notre première méthode [81] et les améliorations que nous y avons apportées depuis, en introduisant les paramètres requis dépendant du type d'écorce afin de l'automatiser. Enfin, nous présentons la classification des écorces pour cette méthode.

2.2.3.1 Le modèle 2D

Le modèle fourni par l'utilisateur est une photographie numérisée en couleur de n'importe quelle écorce, alignée verticalement avec l'axe du tronc ou de la branche d'où

elle provient. Cette image peut être très facilement acquise, soit par un appareil photo numérique, soit grâce à un scanner à plat à partir de tirages photographiques ou de livres. L'échantillon choisi doit être représentatif de l'espèce d'écorce à synthétiser. L'image doit être prise de façon à faciliter la reconnaissance de la structure de l'écorce. Ceci signifie, entre autres conditions, que les faces à moitié éclairées des troncs/branches (avec un éclairage venant de côté au point d'être rasant) doivent être proscrites car les différences entre les parties éclairées et ombragées de l'échantillon perturberaient la méthode qui est basée sur les niveaux de luminosité pour extraire le relief. Une autre raison d'éviter les faces éclairées partiellement est que la lumière pourrait créer des ombres auto-portées trop importantes qui ne représenteraient pas la structure de l'écorce.

D'un autre côté, comme nous en avons discuté au paragraphe 2.2.1, la structure des écorces monochromes est justement donnée par l'illumination venant par le côté. Il faudra donc une lumière ni trop rasante, ni trop diffuse, ni dirigée perpendiculairement à la surface. Pour résumer, un éclairage orienté à environ 45° par rapport à la surface est préconisé, bien que la méthode soit assez tolérante. Ce n'est toutefois pas trop contraignant car, dans le cas où l'utilisateur prendrait lui-même la photo, il suffirait de tourner un peu autour du tronc pour trouver une zone qui convient et, dans le cas d'une photo déjà prise, on peut souvent en découper une partie qui réponde à ce critère, ou bien trouver une autre photo qui se prête mieux à notre méthode.

Il est également recommandé de libérer le modèle du lichen, de la mousse et de la saleté si elle est trop voyante (c'est-à-dire pas seulement concentrée au fond des creux) car ces éléments seraient interprétés comme de l'écorce dans notre traitement alors qu'ils n'en sont pas.

Enfin, l'image doit être découpée de façon à ne conserver qu'une partie qui n'est pas trop déformée par la forme cylindrique de la branche. Idéalement, la texture devra donc être centrée par rapport à l'axe du tronc ou de la branche et ne devra pas être trop large. Cette restriction en largeur est importante mais il est recommandé de maximiser la hauteur autant que possible pour pouvoir réduire la répétition de motifs lors de la synthèse expliquée au prochain paragraphe. La Figure 37 illustre la plupart des contraintes mentionnées ci-dessus.

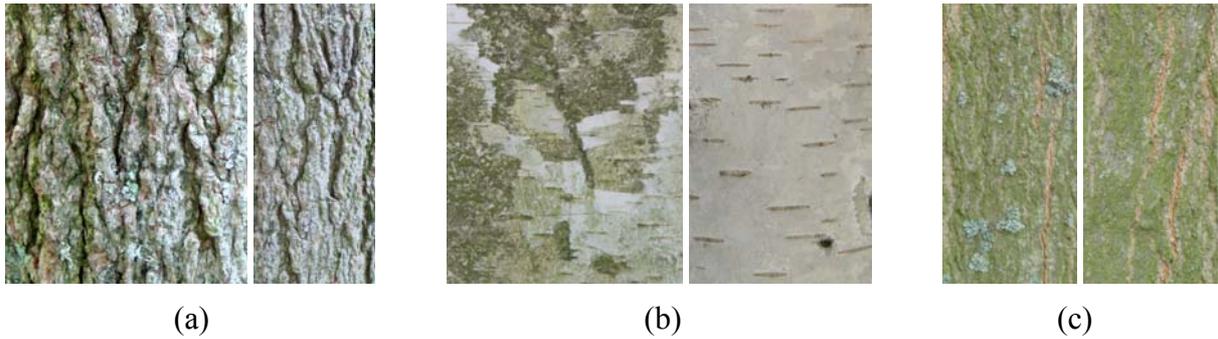


Figure 37 : Trois paires de photos de trois écorces différentes. Chaque paire est composée d'une photographie prise dans de mauvaises conditions (à gauche) et d'une autre correcte (à droite). Les problèmes illustrés ici sont : (a) éclairage partiel ; (b) saleté ; (c) lichen.

Nous verrons au paragraphe 3.3.1 qu'il est possible d'utiliser plus d'une photographie pour représenter les différents états de l'écorce suivant l'âge des branches, pour distinguer le tronc et les branches principales des branches secondaires et des rameaux, par exemple. Néanmoins, toutes ces photographies devront plus ou moins respecter les mêmes règles indiquées dans ce paragraphe. Notons que l'acquisition des textures des rameaux est peut-être plus délicate, compte tenu de leur plus petite taille, mais en contrepartie il est moins nécessaire de faire attention à permettre l'extraction du relief car ces nouvelles branches sont en général assez lisses.

2.2.3.2 Principe de la méthode

D'après notre hypothèse de départ, exprimée au paragraphe 2.2.1, qui montre une corrélation entre les hauteurs et les niveaux de gris du modèle, on peut considérer que les deux niveaux extrêmes qui représentent le fond des craquelures (proches du noir dans le cas général) et les sommets du relief (proches du blanc, en général) caractérisent assez bien la structure de l'écorce et que les niveaux de gris intermédiaires représentent approximativement les hauteurs intermédiaires. Cependant, pour certaines autres écorces, comme le bouleau avec ses très caractéristiques lenticelles et protubérances noires sur fond blanc lisse, les zones claires peuvent être plus profondes que les zones sombres. Dans ce cas, « blanc » signifie « profond » et « noir » signifie « élevé » ; mais les valeurs intermédiaires peuvent toujours être considérées comme des hauteurs intermédiaires. Malgré cette différence avec les autres types d'écorces, on constate toujours deux niveaux

extrêmes assez clairement identifiables. Par conséquent, l'extraction d'une approximation de la hauteur pour ce type d'écorces reste simple puisqu'il suffit de traiter le négatif de l'image.

Cependant, nous avons vu précédemment, au paragraphe 2.2.1, sur les images (b) et (c) de la Figure 36, que l'utilisation directe des niveaux de gris comme carte de hauteur permet, certes, d'obtenir un relief assez approchant mais que ce résultat est néanmoins plus bruité ou chaotique que le modèle. De plus, dans le but d'obtenir des résultats de grande qualité, il est recommandé d'employer des textures en haute résolution ; mais alors, celles-ci sont encore plus susceptibles de contenir du « bruit » (pixels isolés non représentatifs du relief étant contrastés avec leurs voisins) que des images en basse résolution. Notons que ces dernières peuvent également comporter des pixels assez différents de leurs voisins mais, comme ils représentent alors une plus grande surface (du fait de la plus faible résolution), ils sont donc plus représentatifs du relief. La Figure 38 illustre ce problème de bruit pour des écorces d'assez haute résolution. Bien que l'on reconnaisse déjà assez bien la ressemblance entre le modèle et le relief qui en a été directement extrait, on constate que le résultat est trop « bruité » (des pixels isolés ressortent là où l'on s'attend à voir une surface plutôt uniforme), ce qui est particulièrement visible dans le premier exemple mais aussi sur les zones blanches (censées être lisses) du second exemple.

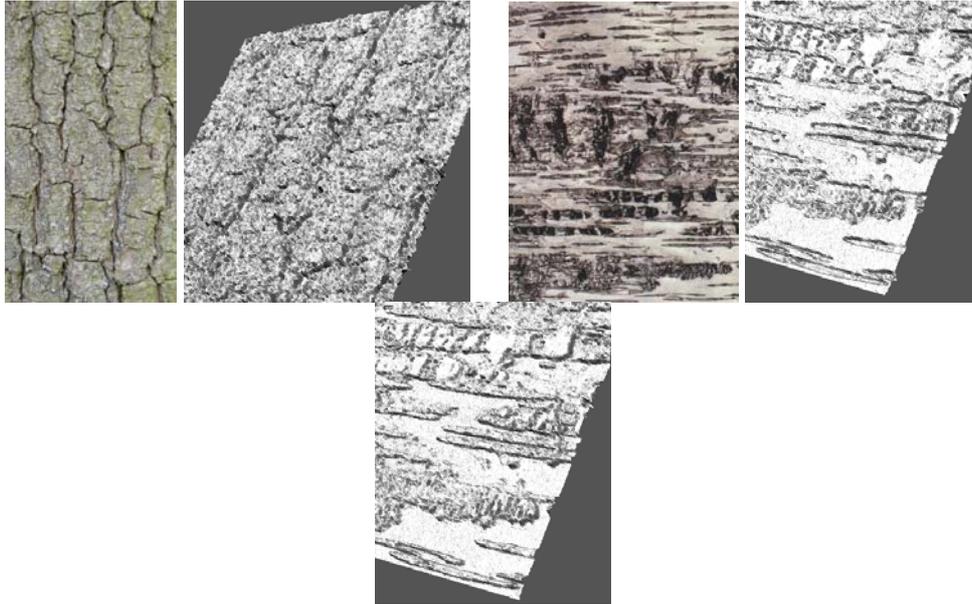


Figure 38 : Deux exemples d'utilisation directe du modèle d'écorce en niveaux de gris comme carte de hauteurs. Les deux reliefs de l'exemple de droite montrent d'abord la carte de hauteurs utilisée telle quelle puis la carte inversée pour convenir à cette écorce de bouleau. Note : les vues monochromes en 3D avec déplacement de surface ont été éclairées avec une lumière venant de la direction en haut à gauche.

Pour éliminer ce « bruit », nous avons expérimenté trois techniques simples pouvant être utilisées seules ou combinées, suivant le type de l'écorce dans notre classification (cf. paragraphe 2.2.3.3). La liste des méthodes de suppression de bruit existantes ne se limite pas aux trois que nous avons retenues et l'on peut en rencontrer beaucoup d'autres dans des logiciels de retouche d'images. Mais les trois techniques suivantes sont assez efficaces et sont très simples à mettre en œuvre. Ce sont donc : le filtrage passe-bas, le filtrage médian et la saturation.

Le filtrage passe-bas

Cette technique, comme par exemple le flou Gaussien, permet de retirer les hautes fréquences et donc les pixels isolés ; mais elle ne convient que pour quelques types d'écorces assez lisses car elle adoucit toute la surface alors qu'il est souvent nécessaire de conserver la rugosité de la texture.

Le filtrage médian

Ce filtrage permet de retirer les pixels isolés. C'est une solution préférable au filtrage passe-bas pour les textures rugueuses car elle préserve mieux les contrastes. Il consiste, pour chaque pixel, à le trier par intensité avec ses huit voisins adjacents (voisinage V8) et à le remplacer par la valeur médiane de la liste triée ($n^{\text{ème}}$ valeur dans une liste de $2n+1$ éléments ou bien moyenne entre la $n^{\text{ème}}$ et la $n+1^{\text{ème}}$ valeur dans une liste de $2n$ éléments, avec n entier strictement positif).

Notons qu'on peut modifier cette méthode pour ne filtrer que les pixels qui diffèrent de la valeur médiane de leurs voisins au-delà d'un certain seuil, afin de supprimer les parasites mais pas les détails. Toutefois, un simple seuil risque d'engendrer une frontière trop nette entre les pixels à traiter et les autres. C'est pourquoi nous interpolons le pixel d'origine avec le pixel filtré lorsque la différence est inférieure au seuil.

Soit $diff$ la différence (en valeur absolue) entre un pixel p et la valeur médiane med de ses voisins, si $diff > seuil$ alors on filtre le pixel normalement (il reçoit la valeur med), sinon le résultat du filtrage médian seuillé de p est donné par l'équation (1) :

$$(1) \quad p_{\text{médian}} = ((seuil - diff)p + (diff \times med)) / seuil$$

La saturation

Cette technique est celle qui préserve le mieux la texture car elle ne retire aucun pixel mais se contente d'en modifier l'intensité. Elle sature certains pixels pour qu'ils se rapprochent du niveau auquel ils sont censés appartenir dans le but d'aplatir plus ou moins les zones devant ne pas être trop chaotiques. Cependant, la différence entre un pixel parasite et ses voisins risque d'être amplifiée si elle était déjà importante au départ. C'est pourquoi, on procèdera d'abord à un filtrage médian seuillé (voir définition ci-dessus) pour éliminer les pixels parasites avant de saturer les niveaux.

Pour les textures où un seul niveau (haut ou bas) est prédominant, comme par exemple le niveau bas de l'écorce de bouleau, on pourra saturer ce niveau au moyen de courbes d'une forme proche de celle utilisée pour la « correction gamma ». Soit $x \in [0;1]$ la

valeur d'un pixel allant du noir au blanc, les équations (2) et (3) des deux courbes de saturation (une pour saturer le niveau bas et l'autre pour le niveau haut) sont :

$$(2) \quad \text{saturation}_{\text{bas}}(x) = x^{\frac{1}{1-\gamma}}$$

$$(3) \quad \text{saturation}_{\text{haut}}(x) = 1 - \text{saturation}_{\text{bas}}(1-x)$$

Le paramètre $\gamma \in [0;1[$ règle le niveau de saturation. La valeur 0 laisse l'image inchangée tandis qu'une valeur proche de 1 saturera complètement le niveau choisi.

Pour les textures à deux niveaux, nous employons une fonction d'augmentation de contraste de forme sigmoïde basée sur les deux fonctions que nous venons de définir. L'équation (4) définissant cette fonction de contraste est :

$$(4) \quad \text{saturation}_{\text{contraste}}(x) = \begin{cases} \frac{\text{saturation}_{\text{bas}}(2x)}{2} & \text{si } x \leq 0,5 \\ \frac{1 + \text{saturation}_{\text{haut}}(2x-1)}{2} & \text{si } x > 0,5 \end{cases}$$

Enfin, pour les quelques textures à trois niveaux, comme par exemple l'écorce de platane qui se décroche en petites plaques dont on peut distinguer trois niveaux, il faut recourir à une saturation qui fait ressortir les niveaux haut et bas ainsi qu'un troisième niveau entre les deux autres. Toutefois, étant donné que ce troisième niveau n'est pas forcément représenté par les pixels proches du niveau de gris moyen (0,5), nous définissons le coefficient m pour représenter le niveau de gris autour duquel devrait être axé le palier intermédiaire du relief. L'équation (5) utilisée pour cela est :

$$(5) \quad \text{saturation}_{3\text{ niveaux}}(x) = \begin{cases} \frac{\text{saturation}_{\text{contraste}}(x/m)}{2} & \text{si } x \leq m \\ \frac{1 + \text{saturation}_{\text{contraste}}((x-m)/(1-m))}{2} & \text{si } x > m \end{cases}$$

La Figure 39 montre l'allure des courbes définies ci-dessus pour une valeur de γ proche de 0,7.

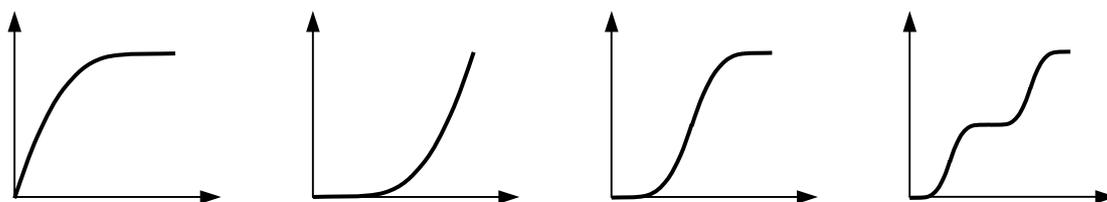


Figure 39 : Fonctions de saturation (valeurs entre 0 et 1 sur les abscisses et les ordonnées). De gauche à droite : niveau haut ; niveau bas ; niveaux haut et bas ; trois niveaux.

Dans un premier temps, nous avons proposé [81] une méthode n'utilisant que la saturation de contraste à deux niveaux ; ce qui permettait déjà de traiter un assez grand nombre de cas. Nous avons alors géré les textures à un seul niveau en déplaçant manuellement le point d'inflexion de la courbe comme dans l'exemple de la Figure 40.

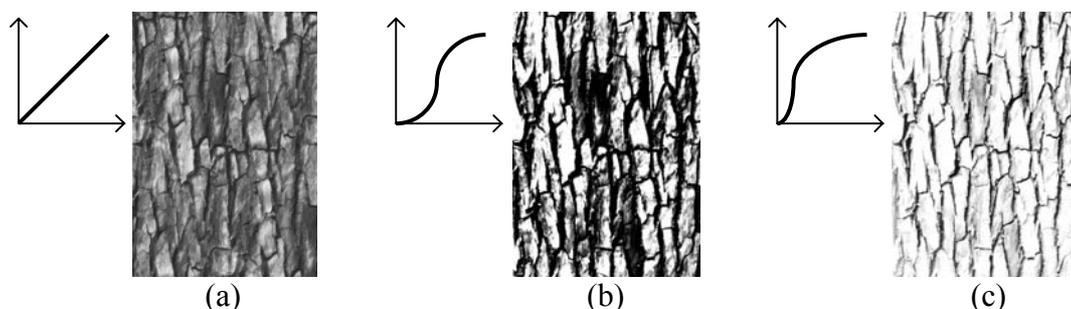


Figure 40 : Séparation des deux principaux niveaux de hauteur de la texture d'écorce (a) par augmentation du contraste, d'abord sans prendre en compte les tâches sombres mais néanmoins à la surface de l'écorce (b), puis en en tenant compte (c).

Dans cet exemple dont le niveau principal est le haut, formé de plaques très plates, la texture contient des zones sombres qui ne doivent pas être prises pour des crevasses. En déplaçant le point d'inflexion de la saturation vers les pixels très sombres, on inclue ainsi un plus grand nombre de pixels assez sombres dans le groupe des pixels clairs et donc élevés. Cependant, l'introduction de la saturation d'un seul niveau à la fois rend le traitement beaucoup plus simple car on constate que la première courbe de la Figure 39 (saturation du niveau haut) ressemble beaucoup à la troisième de la Figure 40 qui était nécessaire auparavant.

Résultats

Couplée à la classification définie au paragraphe suivant et qui permet d'utiliser le(s) traitement(s) le(s) plus adapté(s) pour chaque type d'écorce, cette méthode donne des résultats très acceptables pour la plupart d'entre eux et, même si le relief est souvent approximatif, on reconnaît bien la structure de l'écorce, ce qui contribue à donner l'illusion de réalisme. La Figure 41 montre les exemples utilisés dans la Figure 38 (qui étaient trop bruités) après suppression des pixels parasites.

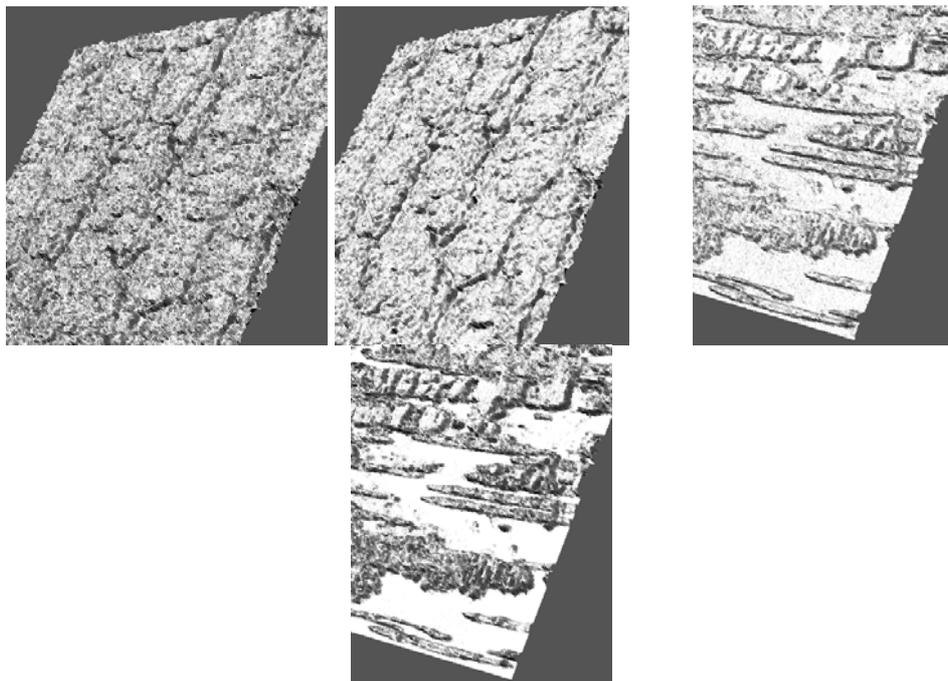


Figure 41 : Les deux exemples de la Figure 38, avant et après réduction du bruit. La texture de gauche a été traitée par filtrage médian tandis que la texture de droite a été traitée par saturation du niveau bas, d'où l'impression de contraste plus fort qui est simplement dû à l'étirement des profondeurs entre les extrema du relief.

Cependant, les écorces qui ne sont pas caractérisées par des niveaux de profondeur clairement identifiables sont un peu moins adaptées, même si elles peuvent toujours être traitées comme des écorces à deux niveaux ou bien simplement par filtrage médian ou passe-bas. Nous verrons aux paragraphes 2.2.4 et 2.2.5 d'autres méthodes pour essayer de les traiter.

Il est possible d'améliorer encore un peu plus les résultats pour les textures dont on sait que leur surface est répartie à peu près équitablement entre les niveaux haut et bas mais dont la luminosité moyenne n'est pas forcément le niveau de gris moyen. Ce peut être le cas par exemple des textures très rugueuses ou chaotiques. Ainsi, grâce à une égalisation d'histogramme [45], on peut rééquilibrer les niveaux des pixels déséquilibrés par la couleur changeante de l'écorce ou par des ombres trop prononcées.

Le principe de ce processus d'égalisation d'histogramme, qui n'est pas un simple étirement d'histogramme, est de faire en sorte que la partie gauche de l'équation (6) suivante tende vers 0,5 (niveau de gris moyen) en ne faisant que déplacer des groupes de pixels d'une même couleur dans l'histogramme (sans changer leur ordre) :

$$(6) \quad \frac{\sum_{i=1}^n i \times \text{histogramme}[i]}{\sum_{i=1}^n \text{histogramme}[i]} \approx 0,5$$

Par exemple, si l'histogramme d'une image de dimensions 8×8 pixels (c.-à-d. 64), en 8 niveaux de gris, est : {10 ; 14 ; 18 ; 12 ; 0 ; 0 ; 10; 0}, alors l'histogramme égalisé deviendra : {10 ; 0 ; 14 ; 0 ; 18 ; 12 ; 0 ; 10}.

Notons enfin qu'une autre caractéristique intéressante de notre méthode permettant d'aplanir des niveaux par saturation est que, dans le cas du niveau bas d'une écorce de bouleau, nous pouvons définir un seuil (proche du niveau bas) qui sépare les zones claires plutôt spéculaires (brillantes) des zones sombres diffuses. Ceci permet d'adapter les propriétés de réflexion de la lumière de la texture aux zones identifiées au cours du rendu.

2.2.3.3 Classification

Le paragraphe précédent offre une boîte à outils permettant d'extraire un relief approximatif mais proche du véritable relief du modèle analysé. Il reste néanmoins à savoir quels outils conviennent le mieux à quel type d'écorce et avec quels paramètres. Afin de simplifier la tâche de l'utilisateur, nous avons donc classifié les différents types d'écorces en quelques catégories représentatives de la plupart des écorces. Il se peut toutefois, vu la grande diversité des écorces et le fait qu'elles sont susceptibles de changer d'aspect au

cours du temps, que ces paramètres nécessitent quelques ajustements mineurs et ils devraient donc plus être présentés à l'utilisateur comme des valeurs par défaut en lui laissant la possibilité de les modifier. Ces outils étant particulièrement intuitifs et rapides, il est tout à fait possible de laisser l'utilisateur les contrôler au cas où notre classification ne s'adapterait pas parfaitement à l'écorce fournie comme modèle.

Dans tous les cas, on effectuera un étirement d'histogramme avant le traitement afin de normaliser les valeurs des pixels dans l'intervalle $[0;1]$, ainsi qu'un filtrage médian avec un seuil égal à 0,25 pour signifier que les pixels qui diffèrent de plus d'un quart par rapport à la valeur médiane de leurs voisins sont très probablement des pics non représentatifs du relief ou bien parasites provenant de grains de poussière ou d'imperfections du mode d'acquisition et nécessitent donc d'être retirés automatiquement. Une fois ces quelques traitements effectués, on procédera au traitement final selon les catégories suivantes (illustrées dans la Figure 42) :

Catégorie 1

Les textures assez rugueuses mais avec des craquelures plus fines que le reste de la surface n'ont pas à recevoir de traitement supplémentaire.

Catégorie 2

Les écorces assez rugueuses et chaotiques dont la transition entre les différents niveaux de hauteur n'est pas représentée par des craquelures franches séparant peuvent être traitées avec une simple égalisation d'histogramme. Cette catégorie regroupe une partie très importante des espèces d'écorces.

Catégorie 3

Les textures lisses doivent être légèrement adoucies au moyen d'un filtrage passe-bas. Nous utilisons un filtrage Gaussien avec un filtre de convolution de dimension de l'ordre de 1% de la largeur de l'image.

Catégorie 4

Cette catégorie regroupe les écorces comportant des crevasses profondes, dont le niveau supérieur est lisse, mais dont certaines zones sombres (mais moins sombres que le fond des crevasses) sont censées être presque aussi élevées que les zones lisses plus claires.

Ces écorces doivent d'abord subir une égalisation d'histogramme pour éclaircir une partie des zones sombres. En effet ces écorces ont tendance à être à la base assez sombres et une égalisation d'histogramme éclaircit donc les niveaux de gris moyens. Enfin, on sature le niveau haut avec un coefficient $\gamma = 0,7$ pour aplanir la surface lisse.

Catégorie 5

Les écorces du type de celle du bouleau dont les zones claires sont lisses et moins élevées que les zones foncées plus rugueuses doivent être inversées (négatif de l'image) puis subir une saturation du niveau bas avec un coefficient $\gamma = 0,7$.

Catégorie 6

Les écorces du type de celle du platane sont formées de couches superposées de plaques relativement planes qui se décollent et laissent apparaître des zones de luminosités différentes suivant le niveau. L'amplitude du relief est faible mais les zones les plus claires sont les plus profondes. On commence donc par inverser l'image pour obtenir son négatif.

Chaque zone étant très plate, nous appliquons une seconde fois le filtre médian. D'éventuelles ombres formées par les plaques les plus élevées peuvent apparaître plus en noir dans le modèle. Afin de les assimiler au même niveau que ces plaques, il faut appliquer une saturation du niveau haut (car les ombres sont blanches après négatif) avec un coefficient $\gamma = 0,5$. Enfin, pour finir d'aplanir les différents niveaux de hauteur dont le nombre peut être estimé à trois, nous les saturons au moyen d'une légère saturation ($\gamma = 0,4$) avec la fonction dédiée aux textures à trois niveaux.

Autres écorces

Parmi les écorces répondant aux principaux critères listés au paragraphe 2.2.3.1, il en reste quelques-unes qui peuvent difficilement être traitées par ces outils. Ce sont

principalement les écorces composées de niveaux de plusieurs couleurs différentes (plus de deux) dont la luminosité ne reflète pas l'élévation. Une autre catégorie est celle des écorces au relief en dent de scie (ou en sillons) comme par exemple l'érable, car sur ce type d'écorce, ce ne sont pas les niveaux haut et bas qui ont le plus d'importance mais les niveaux intermédiaires ; et ceux-ci ont, en général, une luminosité différente suivant que leur normale est orientée ou non vers la source lumineuse. La méthode décrite au paragraphe suivant permet justement de mieux traiter ce genre d'écorces.

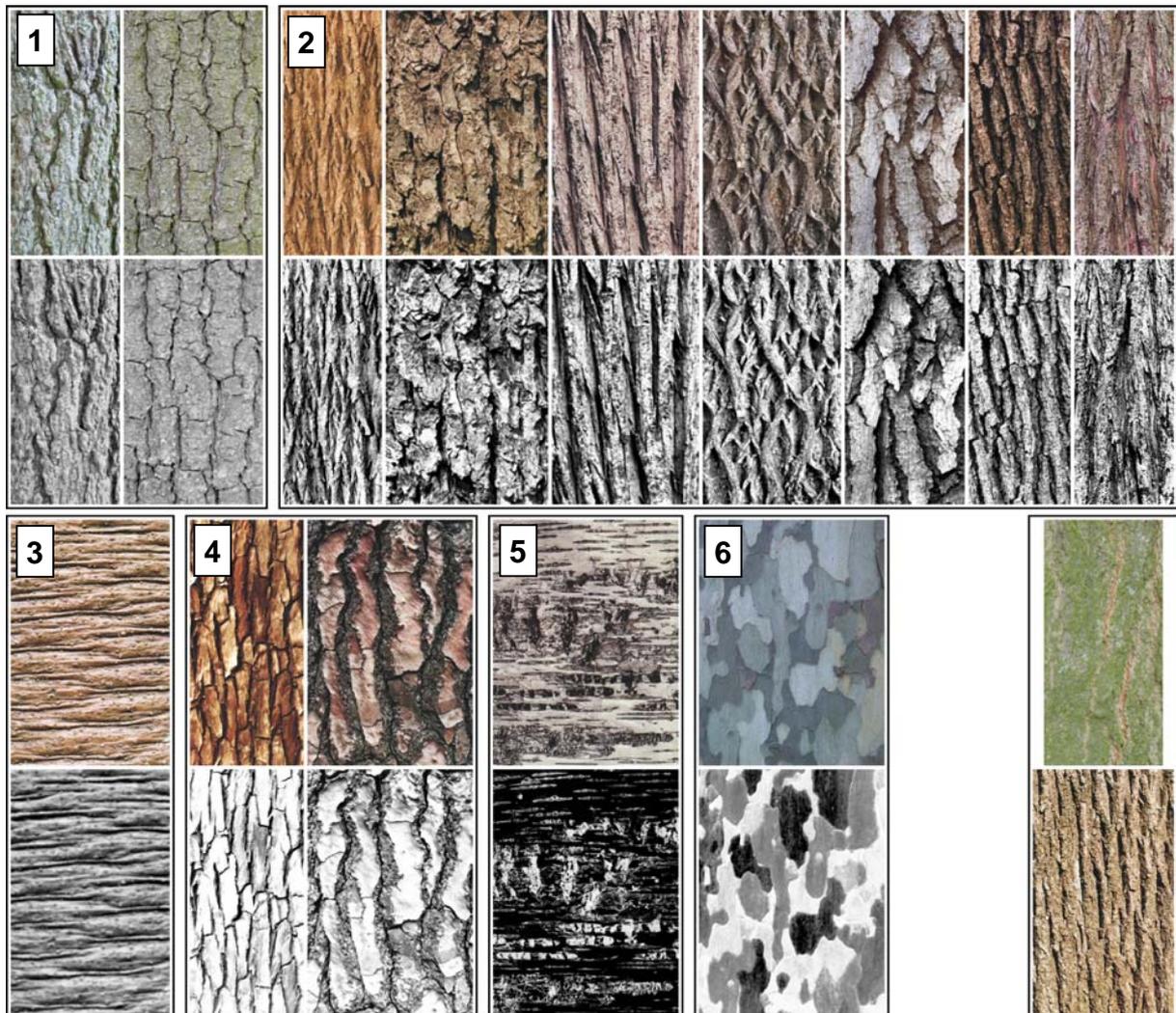


Figure 42 : *Quelques écorces illustrant les différentes catégories de notre classification. Chaque écorce est présentée par couple (modèle et, en dessous, carte de hauteur que nous avons extraite). Chaque cadre englobe une catégorie comportant un ou plusieurs exemples. De gauche à droite et de haut en bas, on trouve les catégories de 1 à 6. Le dernier cadre en bas à droite contient d'autres écorces difficilement traitables par notre méthode.*

2.2.4 Extraction d'un relief cohérent par segmentation d'une image

Nous avons proposé une autre méthode [32] permettant d'extraire le relief correspondant visuellement (mais pas forcément exactement) à une photographie d'une texture fournie comme modèle. La méthode décrite ici est, cette fois-ci, générale et fonctionne avec de nombreux types de textures, dont un grand nombre d'écorces. Elle ne dépend donc pas d'une classification mais requiert en contrepartie une légère interaction avec l'utilisateur. Le principe est d'analyser séparément la macro-texture qui donne l'aspect global de la texture et la micro-texture qui contient les détails, puis de mélanger les deux cartes de reliefs ainsi extraites.

Le paragraphe 2.2.4.1 commence par décrire les conditions que le modèle devra respecter. Nous présentons ensuite la méthode d'extraction de la macro-texture au paragraphe 2.2.4.2, suivie de l'extraction de la micro-texture au paragraphe 2.2.4.3. Puis, nous décrivons la façon de mélanger ces deux couches de relief au paragraphe 2.2.4.4. Enfin, quelques résultats sur des écorces d'arbres sont présentés au paragraphe 2.2.4.5.

2.2.4.1 Conditions à respecter pour le modèle

Tout comme pour la méthode précédente, le modèle doit respecter un certain nombre de règles plus ou moins strictes pour pouvoir être analysé dans les meilleures conditions. Ces contraintes – qui sont globalement les mêmes que celles énoncées précédemment – portent, d'une part, sur la façon dont la photographie doit être prise et, d'autre part, sur la texture elle-même. Ces contraintes sont toutefois moins drastiques que celles imposées par le « *shape from shading* » [57] [140].

La photographie

Les conditions plus ou moins indispensables pour prendre la photographie sont, bien sûr, que la direction de la prise de vue doit être perpendiculaire à la surface et que l'éclairage doit être uniforme sur toute la surface (pas de zones plus éclairées que d'autres) et donc plutôt avec une seule source éloignée de lumière blanche comme le soleil, plus un éclairage ambiant. Ce sont jusqu'ici des conditions que l'on retrouve également dans la méthode précédente et qui permettent d'avoir un modèle non déformé et dont les couleurs ne sont pas faussées.

La texture

Les conditions indispensables au bon fonctionnement de la méthode sont, tout d'abord, que la texture doit être définissable par une carte de hauteurs, donc en $2D\frac{1}{2}$. Ensuite, la surface de support doit être approximativement plane. C'est pourquoi le modèle d'écorce sera relativement étroit pour éviter la courbure de la surface du tronc ou des branches. Enfin, il ne doit pas y avoir de reflets, ce qui n'est de toutes façons pas le cas pour les écorces, même les plus spéculaires.

D'autres conditions, préférables même si elles ne sont pas indispensables, sont que la texture devrait être composée de « bosses » (éléments de relief) facilement identifiables visuellement. D'autre part, ces bosses devraient toutes avoir à peu près la même hauteur et le même profil (forme de la transition entre les creux et le sommet des bosses), si l'on exclue le bruit et les détails. Enfin, il est préférable que la réflectance ne soit pas trop complexe, ce qui n'est pas problématique avec les écorces qui sont en majorité diffuses et, de toutes manières, représentables avec le modèle de Phong [94].

2.2.4.2 La macro-texture

Segmentation

En supposant que les différences de couleur ou de luminosité permettent de différencier les différentes zones d'une texture – et en particulier l'écorce – nous faisons ressortir la structure de la texture d'écorce fournie par l'utilisateur au moyen d'une quantification en un nombre limité de zones de couleurs. Il existe un très grand nombre de méthodes de quantification plus ou moins évoluées, les plus populaires étant la quantification de couleurs par arbre octal [42] et par coupure médiane [53]. Ces méthodes donnent des résultats assez similaires et peuvent donc être utilisées indifféremment pour réduire le nombre de couleurs du modèle.

Dans la plupart des cas, une quantification en deux couleurs est suffisante. Certaines textures peuvent contenir des zones d'ombres un peu plus prononcées que le reste ou bien des reflets spéculaires. Dans ce cas, il pourra s'avérer nécessaire de décomposer l'image en trois, voire quatre, niveaux de couleurs. Ce sera à l'utilisateur d'indiquer s'il souhaite

rajouter un niveau pour les zones plus sombres et/ou pour les zones plus claires (voir discussion sur les images couleur sans ombres ni reflets à la fin de ce paragraphe).

Dans le cas de textures où les niveaux sont représentés par des zones d'une certaine couleur qui a approximativement la même luminosité que le reste de l'écorce, comme par exemple celle de la Figure 43, on pourra éventuellement rajouter encore des niveaux de couleurs si cette couleur spécifique ne ressort pas avec une segmentation en un nombre inférieur de couleurs, même si ce cas particulier reste assez rare. Ainsi, dans cet exemple, il faut augmenter le nombre de couleurs de la quantification jusqu'à ce que ressortent les zones de couleur marron représentant le fond des craquelures.

Il est également possible de différencier les couleurs dans un autre espace que le classique RVB. Les textures avec des zones de couleurs différentes peuvent par exemple être décomposées dans l'espace TSL (teinte, saturation, luminosité) pour segmenter les couleurs suivant leur teinte et leur saturation en ignorant la composante de luminosité. Le principe revient à passer l'image dans l'espace TSL, remplacer la luminosité par une valeur moyenne puis repasser l'image dans l'espace RGB et procéder à la quantification de couleurs.

La Figure 44 illustre ce traitement sur la même texture que dans la Figure 43 et montre qu'il est possible d'obtenir une carte de hauteurs cohérente de plusieurs manières. On constate que les images segmentées servant de base à la macro-texture sont différentes entre ces deux exemples. En revanche, les détails assez bruités du premier exemple qui ne figurent pas dans le second seront rajoutés dans la micro-texture où ils seront mieux gérés, ce qui revient donc à peu près au même au même mais un peu mieux.

Après la quantification, l'utilisateur regroupe manuellement, en quelques clics de souris (au plus, un par couleur résultant de la quantification) – d'une façon assez similaire à celle employée par Premože et al. [97] pour classifier les zones d'un relief alpin – les couleurs qui font partie du même niveau de la macrostructure pour former deux groupes : le fond et les « bosses ». Une interface simple peut ainsi assigner au départ toutes les couleurs à un des deux niveaux (bas, par exemple) et de permettre à l'utilisateur de cliquer sur certaines couleurs pour les assigner à l'autre niveau (haut en l'occurrence). Notons que cette idée de segmentation des zones représentatives de l'écorce a été reprise très

récemment par Wang et al. [131] mais en regroupant des couches de « *textons* », qui sont des éléments représentatifs de la texture à différents niveaux de détails et d'orientation, introduits par Julesz [63] et repris par Malik et al. [79]. Leur méthode nécessite un peu plus de clics mais permet d'effectuer une segmentation un peu plus fine.

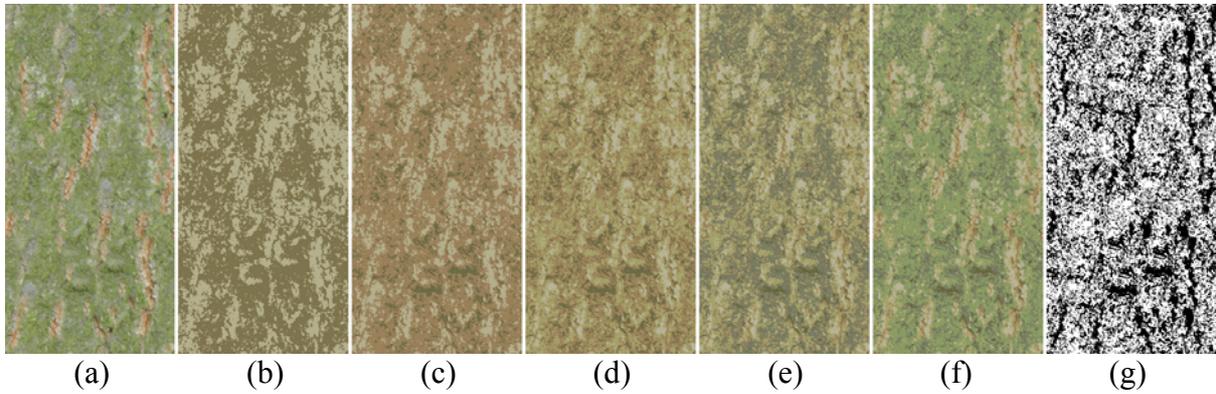


Figure 43 : Segmentation du modèle (a) en plusieurs couleurs pour faire ressortir les couleurs principales que l'on regroupe en deux niveaux (g). Les cinq images du milieu (b–f) représentent, de gauche à droite, une quantification en 2, 3, 4, 5 et 6 couleurs. Dans cet exemple, ce n'est qu'à partir de 5 couleurs qu'on commence à voir ressortir le fond des craquelures de couleur marron et il faut donc quantifier l'image en 6 couleurs pour ne pas retrouver trop de pixels marrons dans le reste de la texture. L'image segmentée (g) a été formée en regroupant manuellement les couleurs de l'image (f) en deux catégories.

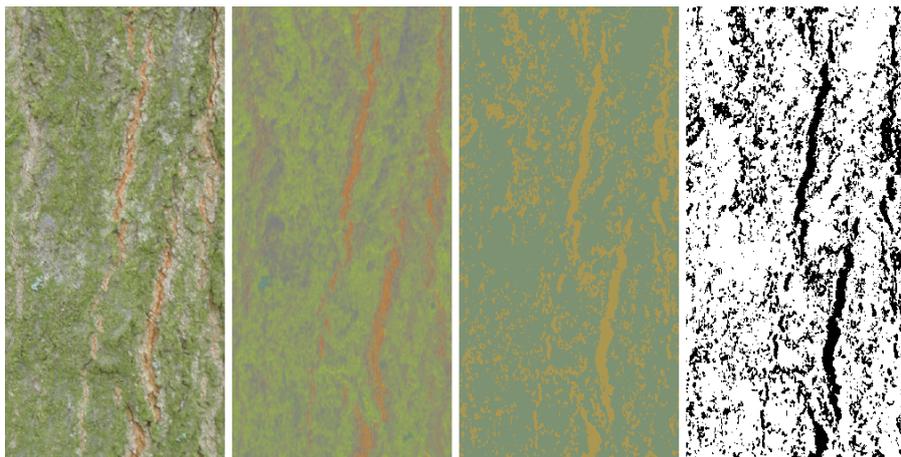


Figure 44 : Segmentation à partir d'une quantification sur la teinte uniquement. De gauche à droite : le modèle ; suppression de la luminosité ; quantification en deux couleurs ; image segmentée en affectant un niveau à chaque couleur.

L'exemple de la Figure 45 montre une écorce légèrement spéculaire qu'il est préférable de segmenter en trois couleurs pour regrouper les deux niveaux les plus clairs ; car, si l'on segmentait en deux niveaux, les reflets blancs « repousseraient » une partie des pixels de la surface dans la catégorie des pixels sombres. Les exemples de la Figure 43 et de la Figure 45 montrent donc différents cas où il peut être nécessaire d'isoler des parties qui se distinguent de la texture et dont on connaît approximativement l'élévation afin de les regrouper avec des zones d'une autre couleur qu'on sait être à peu près du même niveau.

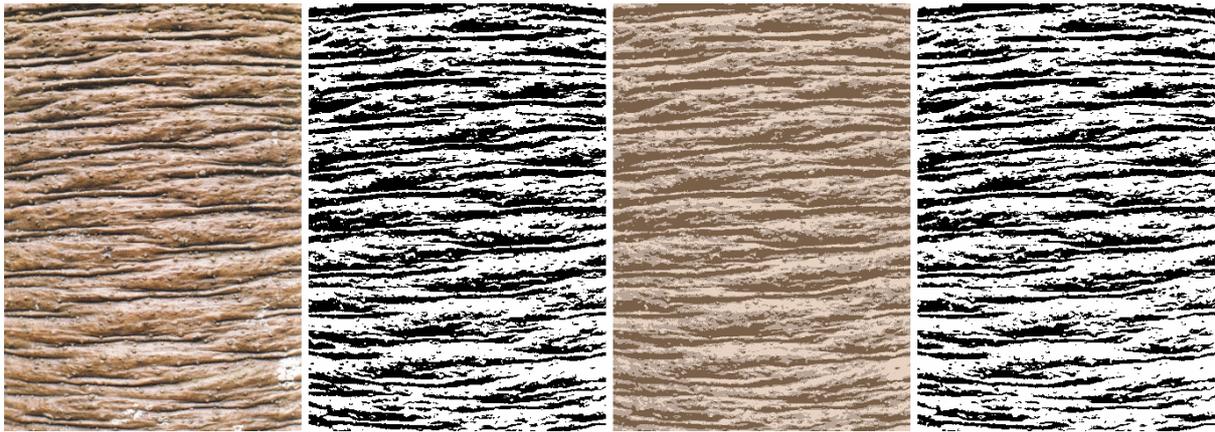


Figure 45 : De gauche à droite : 1°) le modèle ; 2°) segmentation par quantification en deux couleurs ; 3°) quantification en trois couleurs ; 4°) regroupement des deux couleurs les plus claires de la quantification en trois couleurs.

Reconstruction des niveaux intermédiaires

Les niveaux intermédiaires de l'image binaire qui résulte de cette opération sont, ensuite, reconstruits au moyen d'une transformée de distance de chaque pixel par rapport au fond du relief. Les pixels au fond et sur les bords des bosses vaudront 0 et les pixels les plus à l'intérieur des bosses (représentant a priori les sommets les plus élevés) vaudront 1.

Toutefois, avant de procéder à cette transformée, il est nécessaire d'atténuer le bruit en supprimant les petits pixels ou groupes de pixels parasites formant des « trous » dans les bosses. Ceci peut se faire par filtrage médian comme au paragraphe 2.2.3.2 mais on préférera les outils issus du domaine de la Morphologie Mathématique, comme la « fermeture » très efficace pour boucher les petits trous dans le niveau représentant les

bosses. Cette fermeture sera faite au moyen d'un élément structurant de forme arrondie et de petite taille (environ 1% de la largeur de la texture).

Une façon d'effectuer la transformée de distance est de commencer par assigner la valeur 0 à tous les pixels puis de procéder à une érosion morphologique avec un élément structurant formé d'un pixel central et de son voisinage V4 (les quatre pixels situés aux quatre points cardinaux du pixel) ou V8 (les huit pixels qui entourent le pixel central) et d'incrémenter la valeur des pixels qui restent après l'érosion, et ainsi de suite jusqu'à ce qu'il ne reste plus aucun pixel à éroder. L'érosion se ferait en alternant l'élément structurant entre le voisinage V4 et V8 pour en simuler une forme plus arrondie. Ainsi, pour chaque pixel de l'image, si un de ses quatre ou huit voisins (suivant l'étape d'érosion) n'appartient pas à la forme à éroder, c'est que le pixel est proche du bord de celle-ci et doit donc être retiré (remplacé par la couleur du fond). Cette façon de comptabiliser la distance n'est cependant pas très précise. Il est possible mais pas indispensable d'employer une méthode un peu plus précise comme celle de Borgfors [18].

Mais la façon la plus simple et la plus précise, même si c'est la plus lente (ce qui n'est pas un problème ici et c'est pourquoi c'est la solution que nous avons retenue), est encore tout simplement d'assigner à chaque pixel p appartenant à la classe des « bosses » la distance Euclidienne $\min(\text{dist}(p,q))$ pour tout pixel q appartenant à la classe représentant le fond de la texture. Toutefois, pour des textures avec des aplats assez larges formant les bosses, on risque de voir trop nettement la forme de « chapiteau » générée par la transformée de distance. C'est pourquoi, dans ce cas-là, on peut arrêter la progression de la distance à partir d'un certain seuil.

On normalise ensuite cette carte de distances pour avoir des valeurs de 0 à 1. Et on termine enfin par un léger lissage (flou) de la surface pour atténuer la forme linéaire et anguleuse du relief. Il est à noter que cette façon de procéder, que nous avons proposée initialement [32], a tendance à minimiser les bosses par rapport au fond. Il sera donc préférable d'effectuer une dilatation du niveau des bosses (avec le même élément structurant que pour la fermeture) avant de procéder à la transformée de distance. La Figure 46 montre un exemple de macro-texture extraite d'une texture d'écorce.

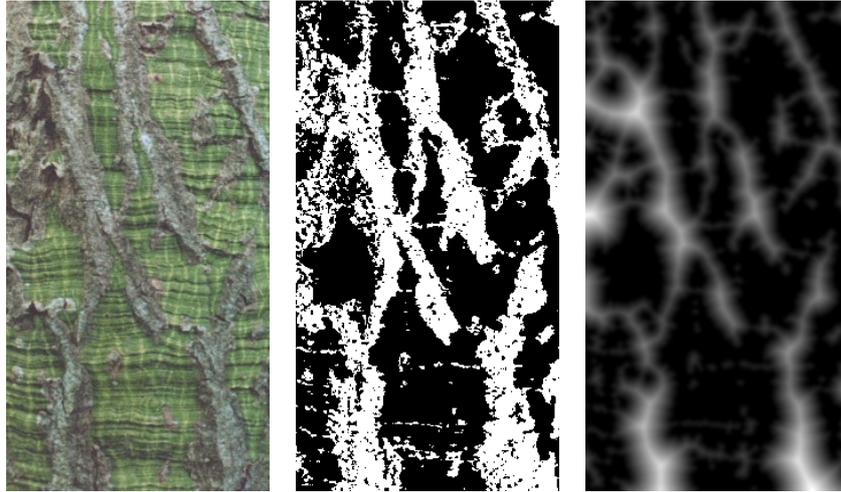


Figure 46 : La texture d'écorce (à gauche) et été segmentée en deux niveaux (au milieu) puis a subi une fermeture, une dilation, une transformée de distance et enfin un lissage (léger flou) pour obtenir la macro-texture (à droite).

Image couleur sans ombres ni reflets

Un problème que l'on peut rencontrer en utilisant l'image couleur du modèle telle- quelle comme carte de couleur est qu'elle contient déjà des informations d'éclairage (ombres et reflets spéculaires) qui feraient donc double emploi et pourraient être contradictoires avec l'éclairage de la scène lors du rendu. Le fait que la notre segmentation puisse détecter ces zones d'ombre et de reflets spéculaires – dans la mesure où l'utilisateur a choisi de gérer l'un de ces cas ou les deux – permet donc de les retirer de la carte de couleur.

La solution consiste donc à remplacer les pixels de l'image couleur, qui appartiennent aux zones marquées comme ombres ou reflets prononcés, par une couleur correspondant à la zone segmentée associée. Dans le cas global, on aura fait une segmentation en quatre couleurs (que l'on peut résumer pour toutes les écorces ou presque par : très foncé, foncé, clair, très clair) où les couleurs extrêmes (très foncé et très clair) seront, chacune, groupées avec les deux niveaux de couleurs intermédiaires adjacents (résultant donc en deux niveaux de couleur : foncé et clair). On remplacera alors les pixels appartenant aux catégories extrêmes de couleurs. Nous avons pour cela envisagé les quatre méthodes suivantes :

- Utilisation directe de la couleur de la zone segmentée. Cette méthode est la plus simple mais présente l'inconvénient que les aplats de couleur risquent de se voir, même si les ombrages rajoutés lors du rendu atténuent ce phénomène.
- Décomposition de la couleur à remplacer en plusieurs composantes : teinte et luminosité (dans l'espace $L^*a^*b^*$ ou XYZ par exemple), puis, remplacement de la luminosité par la luminosité moyenne des pixels de la zone intermédiaire associée.
- Synthèse de texture [4] [7] [35] [36] [54] [95] [133] pour boucher les zones représentant les couleurs extrêmes avec le contenu des zones intermédiaires.
- Connaissant une approximation du relief de la texture, supposée plus ou moins diffuse, on peut considérer, comme Premože et al. [97], que le cosinus de l'angle entre la normale supposée à la surface et la direction estimée de la source lumineuse nous permettrait d'estimer le degré d'illumination en chaque point de la texture. Cependant, il y a trop d'approximations pour espérer obtenir un résultat probant.

Les deuxième et troisième méthodes sont les plus efficaces. Toutefois, comme le montrent nos résultats de la Figure 52, et comme nous l'avons mentionné au paragraphe 2.2.1, certaines zones sombres, qui permettent de bien faire ressortir la structure de l'écorce, ne dépendent pas forcément de l'éclairage et ne doivent donc pas être retirées. Ainsi, dans la majorité des cas on se contentera d'une segmentation en deux couleurs, en conservant le modèle tel-quel comme texture couleur. Les niveaux de couleurs supplémentaires pour supprimer les ombres et reflets ne devront donc être employés que pour des cas particuliers où l'écorce est vraiment monochrome et où les variations de couleur dépendent uniquement de l'éclairage.

2.2.4.3 La micro-texture

Etant donné que la micro-texture représente les petits détails de l'image qui ne font pas partie de la macro-texture que nous avons extraite, l'image que nous traitons pour extraire la micro-texture n'est pas le modèle lui-même mais l'image résultant de la soustraction de la macro-texture au modèle.

Pour effectuer cette soustraction il est nécessaire de connaître la proportion de la micro-texture par rapport à la macro-texture. Cette information ne pouvant à priori pas être

estimée automatiquement à partir d'une simple photographie, ce sera à l'utilisateur d'en donner une valeur approchante (voir Figure 47). Cette proportion sera réutilisée sous une autre forme lors du mélange des deux couches de texture (macro et micro) décrit au paragraphe 2.2.4.4.

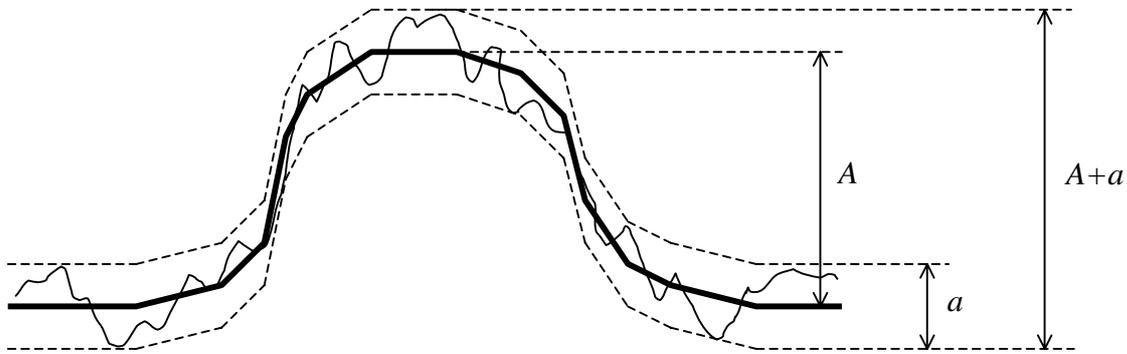


Figure 47 : Estimation de la proportion a/A de la micro-texture (d'amplitude a) par rapport à la macro-texture (d'amplitude A). Dans cet exemple la proportion peut être estimée à $1/3$. La ligne en trait gras représente une vue en coupe d'une bosse de la macro-texture. La ligne continue plus fine et chaotique représente le véritable relief de la texture que l'utilisateur doit estimer visuellement. On note que l'amplitude du relief réel est donc $A+a$.

Soient A et a les amplitudes approximatives respectives de la macro-texture et de la micro-texture, on calculera l'image $T_{détails}$ servant à l'extraction de la micro-texture T_{micro} à partir des deux images nécessaires (le modèle en niveaux de gris $T_{modèle}$ et la macro-texture T_{macro} extraite au paragraphe précédent) « normalisées » en étirant l'histogramme en niveaux de gris allant de 0 à 1. L'équation (7) prenant en compte la proportion a/A est donc :

$$(7) \quad T_{détails} = T_{modèle} - \frac{T_{macro}}{1 + (a/A)}$$

L'image $T_{détails}$ sera à son tour « normalisée » pour que ses niveaux de gris varient de 0 et 1. Et, c'est à partir des observations suivantes que nous traiterons cette image pour retrouver une carte de relief donnant une approximation visuellement satisfaisante de la micro-texture T_{micro} .

Observations

Nous avons remarqué que la taille réduite de la micro-texture fait qu'elle ressemble plus ou moins à du bruit. Or une propriété des textures bruitées est qu'elles ont un aspect similaire si leurs spectres de puissance respectifs, obtenus par transformée de Fourier discrète, ont un aspect similaire.

Nous avons alors cherché une méthode de filtrage d'image qui permettrait d'obtenir une texture dont le spectre de puissance ressemblerait à celui du véritable relief de la texture. Nous sommes donc partis de micro-textures synthétiques afin de connaître exactement leur relief. Nous les avons éclairées avec une source de lumière de direction connue et nous avons comparé les spectres de puissance des cartes de hauteurs connues avec les images éclairées afin de déterminer quel filtrage permettrait de retrouver le relief initial à partir du spectre de puissance de l'image éclairée. La Figure 48 illustre les modèles utilisés.

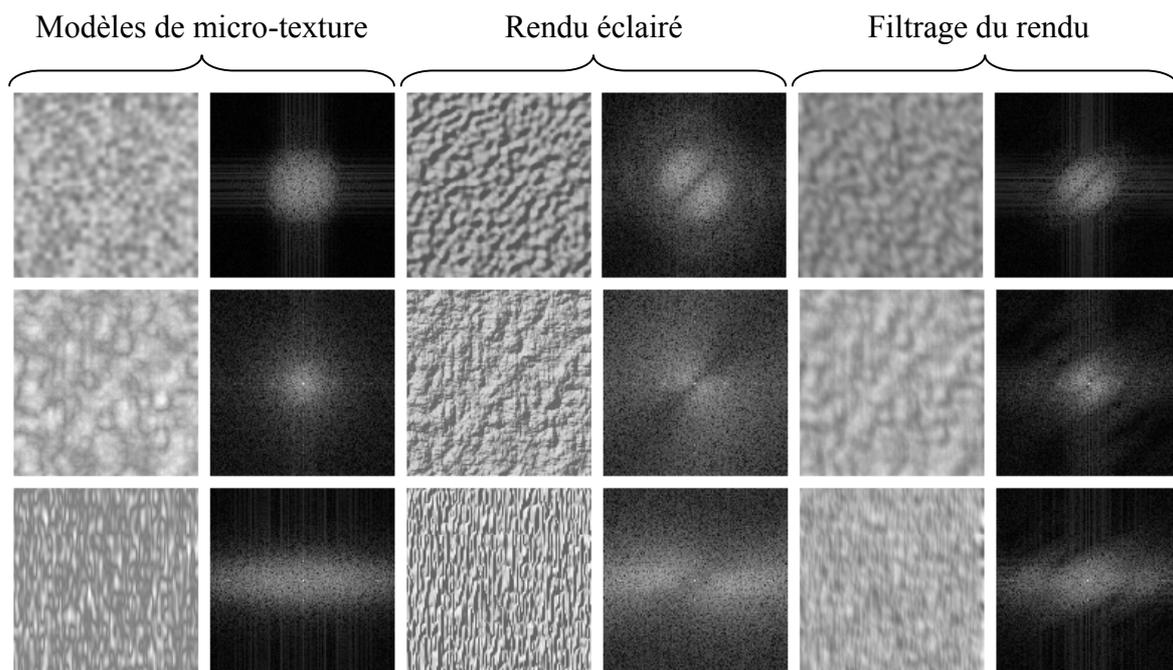


Figure 48 : Etude des spectres de puissance d'une carte de hauteurs et de son relief éclairé (la lumière provient d'en haut à gauche) afin de déterminer quel filtre permettrait de retrouver le relief initial à partir de l'image éclairée. Chaque ligne présente une texture différente. Chaque paire d'images représente une texture et son spectre de puissance obtenu par transformée de Fourier discrète.

Sachant que la luminosité des « pixels » (valeurs discrétisées) des spectres de puissance représente l'amplitude des basses fréquences au centre du spectre et des hautes fréquences sur les bords, nous avons alors pu observer, en comparant ces images dans le domaine fréquentiel, que l'éclairage modifie les spectres de puissance de deux façons : il y a une augmentation des hautes fréquences et une réduction de toutes les fréquences dans la direction perpendiculaire à la direction de la source lumineuse. Il faut donc compenser ces modifications en appliquant l'effet inverse sur l'image éclairée afin d'obtenir un résultat proche des hauteurs réelles de la texture.

La réduction des hautes fréquences se fait par un filtre passe-bas. Le filtrage peut être fait par multiplication dans le domaine fréquentiel, en multipliant le spectre de puissance de l'image par un filtre tel que le filtre Gaussien de la Figure 49, avant de repasser l'image dans le domaine spatial par transformée de Fourier inverse. Mais ce traitement ne fonctionne que sur des images de dimensions carrées. Pour que la méthode fonctionne avec des images de n'importe quelle dimension (nous avons vu qu'il était préférable d'utiliser des images d'écorce hautes mais assez étroites), nous utilisons donc une convolution dans le domaine spatial.

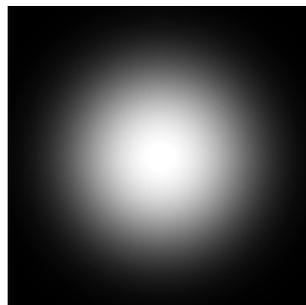


Figure 49 : Filtre Gaussien pour le domaine spectral.

L'augmentation des fréquences dans la direction perpendiculaire à la direction \vec{L} de la source lumineuse se fera au moyen d'un filtrage anisotrope tel que celui de la Figure 50. La largeur des « volants directionnels » (dessinés en petits points sur cette figure) dépend de l'angle α_L la source lumineuse par rapport à la verticale. Si la lumière est rasante, les volants seront étroits ; tandis que si elle vient d'une direction proche du zénith, les volants seront larges au point de ressembler au filtre de la Figure 49. Soit α la moitié de

l'angle formant le volant (celui-ci étant symétrique par rapport au centre du filtre), on calcule donc la formule (8) suivante :

$$(8) \quad \alpha = \frac{\pi}{2} - \alpha_L$$

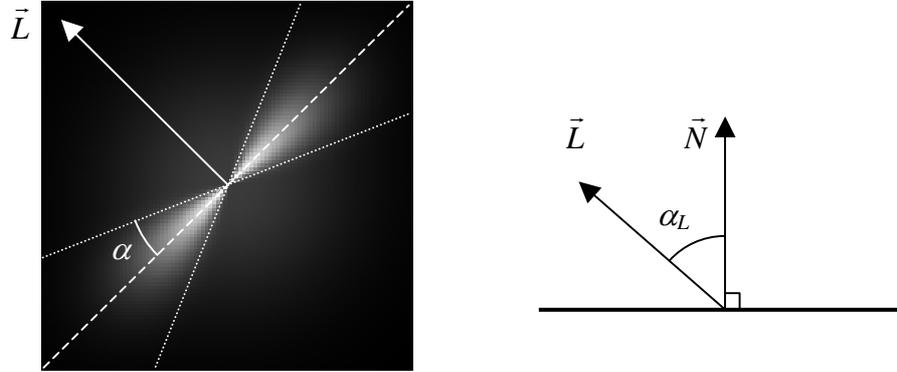


Figure 50 : A gauche : filtre anisotropique perpendiculaire à la direction \vec{L} de la source lumineuse. A droite : α_L est l'angle entre \vec{L} et la normale \vec{N} au plan de base de la texture.

Soit \vec{L}' la projection normée de \vec{L} dans le plan de la texture et, soit \vec{v} le vecteur normé partant du centre du filtre et dirigé vers chaque point (x,y) du filtre dont on doit calculer le coefficient. On commence par déterminer l'angle β que forme \vec{v} par rapport à la direction \vec{L}' avec la formule (9) suivante :

$$(9) \quad \beta = \min(\text{angle}(\vec{v}, \vec{L}'), \pi - \text{angle}(\vec{v}, \vec{L}'))$$

Il s'agit ensuite de savoir si cet angle inclue le point (x,y) dans le volant ou non. Il faut cependant faire une transition douce centrée sur le bord du volant. Cette transition se fera sur un angle $\theta = \min(\alpha, \alpha_L)$. La formule (10) donnant la valeur de chaque point de la partie représentant les volants du filtre est donc :

$$(10) \quad \text{Filtre}_{volants}[x, y] = \begin{cases} 0 & \text{si } \alpha = 0 \\ 1 & \text{si } \alpha_L = 0 \\ \text{lissage}\left(\frac{\beta + \theta - \alpha_L}{\alpha}\right) & \text{si } \theta \neq 0 \end{cases}$$

La fonction de lissage est une sigmoïde définie par l'équation (11) suivante :

$$(11) \quad \text{lissage}(x) = \begin{cases} 1 & \text{si } x \leq 0 \\ \frac{1 + \cos(\pi \times x)}{2} & \text{si } 0 < x < 1 \\ 0 & \text{si } x \geq 1 \end{cases}$$

Enfin, on multiplie le filtre des volants avec le filtre Gaussien de la Figure 49 pour obtenir un filtre (qui devra ensuite être normalisé en divisant chacune de ses valeurs par la somme de toutes les valeurs du filtre) comme ceux de la Figure 51 par la formule (12) suivante :

$$(12) \quad \text{Filtre}[x, y] = \text{Filtre}_{\text{volants}}[x, y] \times e^{\frac{-(x^2+y^2)}{2,5 \times \text{Taille}_{\text{Filtre}}}}$$

Dans la formule ci-dessus $\text{Taille}_{\text{Filtre}}$ est le côté du filtre carré discrétisé qui sera utilisé et 2,5 est un coefficient constant tel que le résultat soit négligeable sur les bords du filtre afin d'en faire un filtre à réponse impulsionnelle finie (« *finite impulse response filter* »).

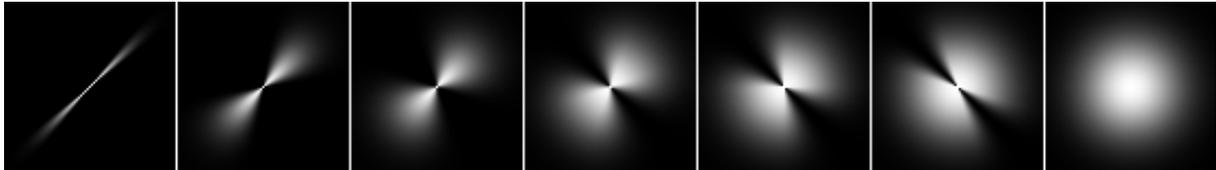


Figure 51 : Série de filtres pour des angles α_L décroissant de $\pi/2 - \varepsilon$ (lumière rasante) à $0 + \varepsilon$ (lumière proche du zénith).

Etant donné qu'il est nécessaire de faire une convolution dans le domaine spatial plutôt qu'une multiplication dans le domaine fréquentiel, afin de pouvoir traiter des images rectangulaires, nous transformons notre filtre dans le domaine spatial par transformée de Fourier inverse puis nous lui donnons une taille à l'échelle des détails à traiter, c'est à dire en pratique une taille d'environ 5x5 pixels, allant de 3x3 pour de petites textures à 7x7 pour des textures en haute résolution.

2.2.4.4 Mélange des deux couches de relief

Le mélange des couches de la macro-texture avec la micro-texture (toutes deux « normalisées » avec des niveaux de 0 à 1) est fait à l'aide d'un coefficient $C_{mélange}$ défini par l'utilisateur. Nous avons déjà eu recours à un paramètre similaire sous la forme d'un rapport a/A au paragraphe 2.2.4.3 consacré à la micro-texture où A et a sont les amplitudes respectives de la macro-texture et de la micro-texture, dont la somme $A+a$ vaut 1, afin d'obtenir un relief final normalisé avec des niveaux allant de 0 à 1. Le coefficient $C_{mélange}$ est donc simplement l'amplitude a que nous avons renommée pour mieux convenir à l'utilisation que nous en faisons ici. Donc, si l'on note ρ le rapport $a/A = C_{mélange}/A$, on a $C_{mélange} = \rho A$. Or, $A = 1 - C_{mélange}$, donc $C_{mélange} = \rho(1 - C_{mélange})$, d'où l'équation (13) :

$$(13) \quad C_{mélange} = \frac{\rho}{1 + \rho}$$

Avec cette nouvelle représentation, $C_{mélange} = 0$ signifie que la micro-texture est ignorée ; $C_{mélange} = 1$ signifie que c'est la macro-texture qui est ignorée et $C_{mélange} = 0,5$ donne la même importance aux deux couches. La formule (14) de mélange des deux couches *Macro* et *Micro* en une carte de hauteurs *Relief* est donc :

$$(14) \quad Relief = (C_{mélange} \times Micro) + ((1 - C_{mélange}) \times Macro)$$

2.2.4.5 Résultats

La Figure 52 illustre deux exemples d'arbres rendus avec des écorces analysées par notre méthode. Comme pour une grande majorité des écorces, nous avons utilisé le modèle tel-quel pour la couche de couleur, plutôt qu'une version où les ombrages auraient été retirés, car les différences de luminosité des modèles employés ne sont pas seulement dues à l'éclairage mais aussi et surtout à la couleur de la texture.

La transformée de distance du premier exemple de cette figure a été seuillée car le macro-relief de cette texture est relativement plat, tandis qu'il n'y a pas eu de seuillage pour le deuxième exemple car la macro-texture de cette écorce a plus ou moins une forme de « chapiteau » formé de crêtes.



Figure 52 : Deux exemples d'arbres dont le relief de la texture d'écorce a été extrait grâce à notre méthode (le modèle et le relief sont en haut à droite de chaque exemple).

2.2.5 Une méthode combinant les deux précédentes

La méthode [32] définie au paragraphe précédent distingue la macro-texture (zones représentant la structure principale du modèle) déterminée par segmentation et la micro-texture (petits détails) analysée à partir de la différence entre l'image en niveaux de gris et la macro-texture. Cette idée de séparer ces deux niveaux d'échelle nous paraît intéressante mais il est clair qu'une meilleure connaissance de la texture fournie comme modèle améliorerait l'interpolation des zones identifiées par la segmentation en évitant de supposer systématiquement que le profil est une simple pente lissée. Cela permettrait également d'automatiser une grande partie du traitement. Par conséquent, nous proposons une extension de cette méthode en ce qui concerne l'analyse de la macro-texture, tout en conservant l'extraction de la micro-texture telle qu'elle a été définie dans la méthode précédente.

Afin d'apporter des informations supplémentaires sur la façon dont les principaux niveaux de hauteur de la texture doivent être interpolés, nous procéderons donc de la même manière que pour la première méthode décrite au paragraphe 2.2.3 : c'est-à-dire à l'aide d'une classification où nous aurons défini des paramètres pour chaque type d'écorce identifié. La notion de type d'écorce est toujours la même que dans la première méthode mais les catégories sont différentes.

Globalement, notre nouvelle méthode détermine la macro-texture par segmentation (réduction de couleurs puis regroupement des couleurs assimilables à un même niveau d'élévation) puis elle interpole les niveaux de hauteur entre ces zones. Enfin, elle mélange la couche de macro-texture avec la micro-texture extraite de la même manière qu'avec la méthode décrite au paragraphe 2.2.4.3.

2.2.5.1 Retrouver la macro-texture

Comme dans la méthode précédente (cf. paragraphe 2.2.4), notre analyse commence par la segmentation de l'image fournie comme modèle en un petit groupe de classes de pixels qui sont supposées représenter approximativement les principaux niveaux d'élévation. Par exemple, le fond des craquelures et la surface de l'écorce sont les deux principaux niveaux identifiés pour de nombreuses écorces. Cependant, contrairement à la

méthode précédente, nous pourrions également traiter les écorces pour lesquelles on distingue plus de deux niveaux (voir le deuxième exemple de la Figure 53) ; mais le principe reste le même. Notons toutefois que le nombre de couleurs résultant de la quantification (de même que la hauteur de chaque niveau, ainsi que d'autres paramètres qui serviront à mieux reconstruire le relief intermédiaire entre les niveaux segmentés) n'est pas donné directement par l'utilisateur mais par le type de l'écorce qu'il indiquera en se basant sur la classification présentée au paragraphe 2.2.5.4.

La quantification laisse des pixels parasites qu'il n'est toutefois pas nécessaire de retirer (à l'aide d'un filtrage de réduction de bruit) car ils apportent souvent des détails et seront atténués lors de la reconstruction des niveaux intermédiaires, même pour les écorces dont le type indique que la transition d'un niveau à l'autre est abrupte. Enfin, le type d'écorce permet également de savoir si la texture est susceptible de contenir des zones spéculaires qui devraient donc être quantifiées à part et regroupées avec la seconde couleur la plus claire ou, au contraire, des zones d'ombre qui devraient être regroupées avec la seconde couleur la plus sombre, comme il en est question au paragraphe 2.2.4.2.



Figure 53 : Deux exemples de segmentation en zones représentatives de la macro-texture. La texture d'écorce du deuxième exemple nécessite une segmentation en trois classes.

Outre l'utilisation d'une classification (afin d'automatiser un maximum de réglages), permise par le fait que nous nous restreignons à ne traiter que des écorces (contrairement à la méthode décrite au paragraphe 2.2.4.2 qui se veut générale), la principale différence avec la méthode précédente intervient dans la reconstruction des niveaux intermédiaires entre les principaux niveaux, détaillée ci-dessous.

2.2.5.2 Reconstruction des niveaux intermédiaires de la macro-texture

La segmentation est une approximation de la carte de hauteur puisque chaque zone segmentée a sa propre élévation moyenne, donnée par le type de l'écorce. Mais les niveaux intermédiaires doivent être reconstruits par une interpolation qui dépend, une fois encore du type de l'écorce. En effet, il y a, par exemple, des écorces avec de profondes craquelures abruptes qui ne nécessitent presque pas d'interpolation, tandis que d'autres sont plus lisses et nécessitent une transition en forme de sigmoïde (comme une courbe de Gauss) ou bien une simple interpolation linéaire, etc.

Cette interpolation n'est plus faite par transformée de distance mais au moyen d'un filtre de convolution appliqué sur la texture segmentée. Les coefficients de ce filtre décroissent linéairement du centre vers les bords. En outre, on peut noter pour la plupart des écorces une orientation de la texture, soit verticalement, soit horizontalement, ce qui implique que la transition verticale d'un niveau de hauteur à un autre peut différer de la transition horizontale. Par conséquent, le filtre n'est pas nécessairement isotropique mais peut être allongé verticalement ou horizontalement, selon l'orientation de la texture. Cette orientation est limitée à ces deux directions, ce qui simplifie la détermination de la forme du filtre que nous calculons automatiquement. La Figure 54 montre un filtre pour une texture sans orientation particulière puis pour une texture plutôt horizontale.

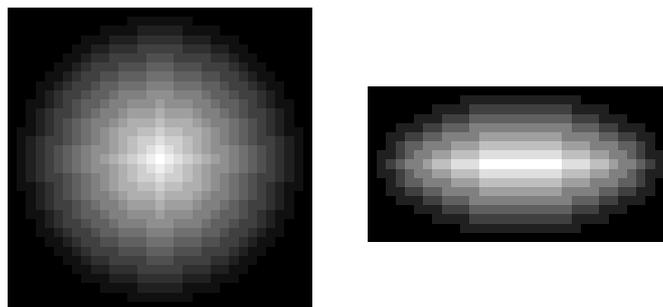


Figure 54 : Filtres linéaires pour une image sans orientation de texture (à gauche) et avec orientation horizontale (à droite).

Le rapport $ratio_{texture}$ entre la largeur et la hauteur du filtre (L_{filtre}/H_{filtre}) est calculé grâce à une analyse spectrale (transformée de Fourier discrète) d'une partie carrée de la texture, prise au milieu du modèle (ou de l'image segmentée car les résultats sont très

proches) et la plus large possible (le modèle étant généralement plus haut que large, le côté du carré sera donc en général la largeur du modèle), pour produire un spectre de puissance de côté $taille_{SP}$ (voir image Figure 55.c). En fait, nous voulons comparer les amplitudes « verticales » avec les amplitudes « horizontales ». Ainsi, dans le spectre de puissance, la somme de toutes les amplitudes pondérées, chacune, par la valeur absolue du cosinus de l'angle formé par le pixel correspondant, permet d'obtenir une valeur représentant l'influence des amplitudes par rapport à l'axe des abscisses (partie supérieure de l'équation (15) ci-dessous). Respectivement, la valeur représentant l'influence des amplitudes par rapport à l'axe des ordonnées (partie inférieure de l'équation (15) ci-dessous) est calculée en remplaçant le cosinus par le sinus de l'angle. Le rapport $ratio_{texture}$ entre ces deux valeurs, indiquant l'orientation de la texture est donc :

$$(15) \quad ratio_{texture} = \frac{\sum_{i=0}^{taille_{SP}-1} \sum_{j=0}^{taille_{SP}-1} \frac{Ampl(i, j) \times |C(j)|}{\sqrt{C(i)^2 + C(j)^2}}}{\sum_{i=0}^{taille_{SP}-1} \sum_{j=0}^{taille_{SP}-1} \frac{Ampl(i, j) \times |C(i)|}{\sqrt{C(i)^2 + C(j)^2}}}$$

$Ampl(i, j)$ donne l'amplitude du spectre de puissance aux coordonnées (i, j) . Les cosinus et sinus ne figurent pas directement dans l'équation car ils sont calculés à partir des coordonnées (i, j) transformées par la fonction C définie par l'équation (16) de sorte qu'une coordonnée (respectivement i en abscisse et j en ordonnée) soit traduite dans le repère centré au milieu du spectre de puissance ; d'où :

$$(16) \quad C(x) = x - \frac{taille_{SP} - 1}{2}$$

Si l'on connaît l'aire $aire_{filtre}$ du filtre, donnée sous la forme d'un pourcentage de l'aire du morceau carré analysé et dépendant du type de l'écorce, on peut alors calculer les dimensions du filtre, données par les équations (17) et (18) :

$$(17) \quad L_{filtre} = \sqrt{aire_{filtre} \times ratio_{filtre}}$$

$$(18) \quad H_{filtre} = \sqrt{\frac{aire_{filtre}}{ratio_{filtre}}}$$

La Figure 55 montre un exemple d'écorce (a) de direction horizontale assez prononcée que nous avons segmentée (b) puis transformée en spectre de puissance (c) pour en déduire le ratio du filtre afin de filtrer l'image segmentée pour donner le résultat (d). Il est à noter que l'aspect horizontal de la texture se traduit par un spectre de puissance orienté verticalement.

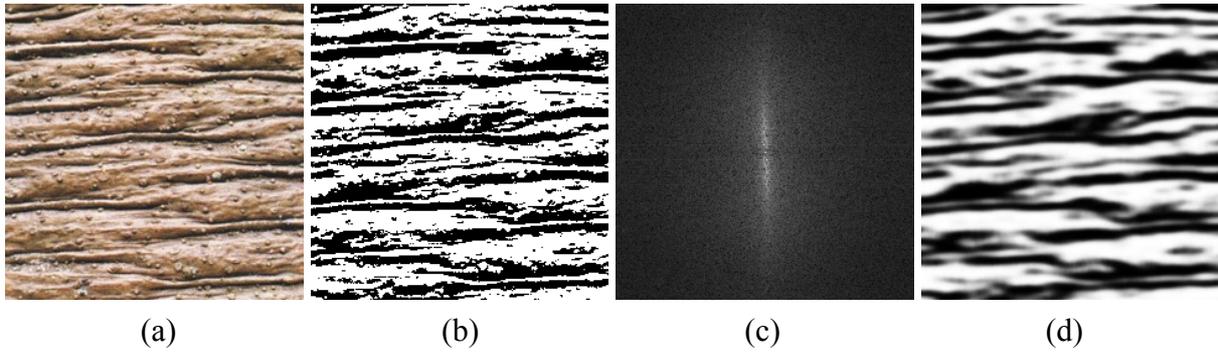


Figure 55 : a) modèle ; b) segmentation correspondante en deux niveaux ; c) spectre de puissance (obtenu par transformée de Fourier discrète) donnant les dimensions du masque (le ratio calculé ici vaut environ 3, ce qui signifie que la texture est très orientée horizontalement) ; d) filtrage de l'image segmentée, avec le masque de convolution calculé.

L'image résultante contient donc des pixels ayant des valeurs allant de 0 à 1. Il ne reste alors plus qu'à la convertir à l'aide d'une courbe de profil pour régler la hauteur des pixels intermédiaires. La Figure 56 montre les courbes de profil que nous utilisons.

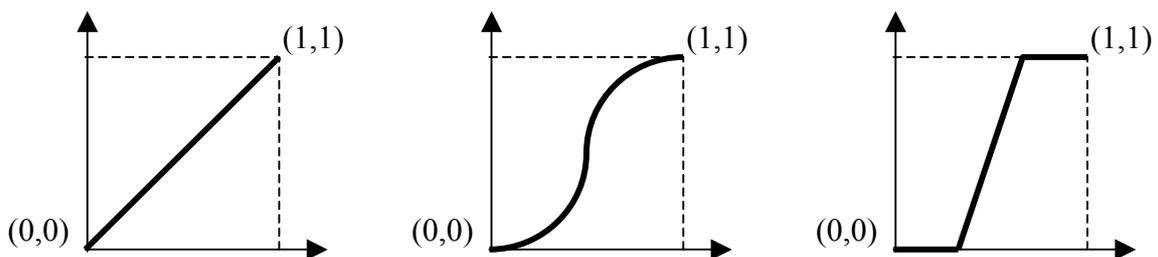


Figure 56 : Courbes de profil dépendant du type d'écorce. A gauche : identité (on ne modifie pas le profil). Au centre : sigmoïde (permet d'obtenir un profil doux). A droite : transition abrupte (permet de représenter des transitions abruptes tout en ayant réduit les pixels parasites grâce au filtrage par convolution).

2.2.5.3 Ajout de détails

Les détails perdus lors de la segmentation et qui n'ont pas été récupérés par l'interpolation sont extraits exactement de la même manière qu'avec la méthode décrite au paragraphe 2.2.4.3. Il en va de même pour le mélange des couches de macro-texture et micro-texture, mais le coefficient de mélange n'a pas besoin d'être indiqué par l'utilisateur car il sera donné par le type de la texture.

2.2.5.4 Classification

Tout comme pour la première méthode proposée (cf. paragraphe 2.2.3) nous avons établi une classification, élaborée empiriquement à partir d'essais sur plusieurs modèles d'écorces similaires, permettant de traiter de la même façon un certain nombre d'écorces plus ou moins similaires. Bien qu'il y ait quelques points communs entre certaines catégories de cette méthode et de la première classification, il ne faut cependant pas les confondre car il y a des différences importantes. Les catégories ci-dessous n'englobent pas la totalité des espèces d'écorces existantes. Celles qui ne figurent pas dans cette classification peuvent être gérées par une des deux méthodes précédentes ou bien ne répondent pas aux conditions requises pour le modèles par l'ensemble des trois méthodes.

Catégorie 1

La première catégorie englobe les textures d'écorce relativement plates avec seulement une micro-texture. Le coefficient $C_{mélange}$ vaut donc 100% et il est inutile d'extraire la macro-texture.

Catégorie 2

Cette catégorie regroupe les textures très rugueuses. Nous avons constaté, pour la première méthode au paragraphe 2.2.3, qu'il était préférable d'égaliser l'histogramme pour rééquilibrer la proportion de pixels élevés et profonds. Etant donné que la quantification de couleurs dépend beaucoup des différences entre les pixels, il est important d'égaliser l'histogramme avant de procéder à la quantification en deux couleurs nécessaire à l'isolement des deux principaux niveaux de cette catégorie.

L'interpolation entre ses deux niveaux se fera avec un filtre linéaire (cf. paragraphe 2.2.5.2) d'aire représentant environ 1% de la surface totale du modèle. Il n'y aura pas besoin ici de transformation des niveaux intermédiaires par courbe de profil. Enfin, les deux couches de macro-texture et de micro-texture ont sensiblement la même importance. Le coefficient $C_{mélange}$ vaut donc 50%.

Catégorie 3 :

Cette catégorie est constituée des texture d'écorces lisses (et spéculaires). La segmentation en deux niveaux est donc faite à partir d'une quantification en trois niveaux de gris dont ont regroupe les deux niveaux les plus clairs pour former le niveau élevé.

Ces textures étant très lisse, l'aire du filtre recréant une transition entre les deux niveaux est assez large et représente 5% de la surface du modèle. D'autre part, le relief de la micro-texture est assez faible et $C_{mélange}$ vaut environ 25%. Enfin, afin de respecter la douceur du relief, nous ajustons le profil au moyen d'une courbe sigmoïde.

Catégorie 4

Nous avons regroupé dans cette catégorie les textures rugueuses mais peu élevées dont le relief dépend surtout de la couleur. Elles sont quantifiées/segmentées en deux niveaux par teinte. Il est toutefois impossible de savoir à l'avance laquelle des deux couleurs résultant de la quantification sera le niveau bas ou haut. En supposant, que le nombre de pixels occupés par les crevasses est inférieur au nombre de pixels du niveau élevé sur ce genre de texture (ce qui est vrai pour celle de la Figure 44, par exemple) il est alors possible d'estimer à quel niveau correspond chaque couleur de la quantification. Il est cependant préférable de laisser la possibilité à l'utilisateur d'inverser ces niveaux dans les rares cas où cette règle ne se vérifierait pas.

L'aire du filtre de convolution est moyenne et vaut 1% de la surface totale du modèle. Les deux niveaux de détail de la texture sont d'importance équivalente et $C_{mélange}$ vaut donc 50%. Enfin, la texture n'ayant pas un relief doux, le profil reste inchangé.

Catégorie 5

Cette catégorie regroupe les écorces à lenticelles (et protubérances) comme le bouleau. La segmentation se fait en deux niveaux suivant la luminosité en transformant tous les pixels gris foncé (d'intensité inférieure à 0,5) en noir et les autres en blanc.

Un peu comme pour la quatrième catégorie, les lenticelles peuvent être plus sombres ou plus claires que le fond suivant l'espèce d'écorce et, afin de limiter le nombre de catégories, il est intéressant de regrouper toutes les écorces à lenticelles et de déterminer automatiquement si le relief doit être inversé (lenticelles et protubérances sombres) ou pas. Les lenticelles occupent en règle générale moins de surface que le fond de la texture. On inverse donc les niveaux si le nombre de pixels blancs est supérieur au nombre de pixels noirs. Cette condition étant toutefois incertaine, il vaut mieux laisser la possibilité à l'utilisateur d'inverser manuellement l'image si nécessaire.

L'aire du filtre de convolution est moyenne, afin de ne pas trop élargir les lenticelles, mais assez large pour ne pas trop aplatir le niveau haut qui doit être assez bruité et surtout pour abaisser le niveau des fines lenticelles dont le relief est beaucoup moins prononcé que les parties les plus rugueuses. Cette aire vaut donc 1% de la surface du modèle.

Contrairement aux autres types d'écorce, il faut faire attention au bruit qui peut être présent sur le modèle au niveau bas qui est censé être lisse. La macro-texture devrait être suffisamment lisse mais la micro-texture ne le sera probablement pas. Nous faisons donc varier le coefficient $C_{mélange}$ (qui doit être de 50% pour le niveau haut) pour chaque pixel en fonction du niveau d'élévation grâce à l'équation (19).

$$(19) \quad C_{mélange}(x, y) = 50\% \times Macro(x, y)$$

Enfin, la courbe de profil est une sigmoïde afin d'adoucir les bords des lenticelles. La rugosité des protubérances sera restituée par la micro-texture.

Catégorie 6

Cette catégorie est constituée des textures faites de fines plaques à environ trois niveaux, telles que l'écorce de platane. Tout comme la catégorie de même numéro dans la classification de la première méthode (cf. paragraphe 2.2.3.3) l'image est d'abord inversée

(négatif) mais avant, plutôt que de saturer les petites ombres pour les mélanger avec les zones sombres, on effectue une quantification en quatre niveaux de luminosité et on regroupe les deux plus sombres (qui deviendront les deux plus clairs après inversion). La texture est donc finalement segmentée en trois niveaux de hauteur.

L'aire du filtre doit être très faible (0,1% de la surface du modèle) mais pas trop car il faut pouvoir atténuer les pixels parasites restant de la quantification. La courbe de profil sera donc abrupte (courbe de droite de la Figure 56) afin de représenter le profil souhaité. Enfin, le relief de la micro-texture est assez faible et $C_{mélange}$ vaut 25%.

Catégorie 7

Les textures chaotiques aux bords peu rugueux sont plusieurs points communs avec celles de la seconde catégorie : elles nécessitent une égalisation d'histogramme puis quantification/segmentation en deux niveaux par rapport à la luminosité.

Etant donné qu'elles sont un peu plus lisses, l'aire du filtre fait 2% de la surface du modèle. La micro-texture est un peu moins importante, avec $C_{mélange}$ à 40% ; et le profil des bosses suit une courbe sigmoïde.

2.2.6 Synthèse des trois méthodes proposées

La première méthode, présentée au paragraphe 2.2.3, fonctionne dans le cas général où le niveau d'illumination de chaque pixel du modèle peut être assimilé à son élévation au moyen de quelques traitements d'images simples. Dans cette hypothèse, c'est la méthode qui donne les meilleurs résultats car c'est celle qui altère a priori le moins la texture originale.

Cependant, il y a un certain nombre de textures qu'elle ne gère pas très bien, comme par exemple les écorces avec un relief anguleux en forme de chapiteaux (l'écorce d'érable en est un exemple) ou encore les écorces où l'élévation dépend plus de la couleur que de la luminosité ou bien où la micro-texture est très dépendante de l'éclairage. Dans ces cas-là, la méthode la plus appropriée est la seconde méthode, décrite au paragraphe 2.2.4. La position du sommet des chapiteaux ne correspond pas toujours exactement à l'endroit où ils sont sur le modèle mais la structure générale de l'écorce est respectée ce qui permet

d'obtenir tout de même des résultats visuellement réalistes. Cette méthode fonctionne également avec d'autres types d'écorces (et même avec d'autres textures) mais, elle ne donne pas de résultats particulièrement meilleurs que les autres méthodes décrites dans ce chapitre et qui ont l'avantage d'être plus automatisées.

Enfin, nous avons proposé une troisième méthode, au paragraphe 2.2.5, qui reprend la notion de connaissance a priori de l'aspect du relief à l'aide d'une classification et qui a été utilisée dans la première méthode. Toutefois, la classification est différente car il ne s'agit plus de modifier une image existante supposée ressembler déjà beaucoup au résultat souhaité mais, cette fois-ci, de reconstruire le relief à partir de la structure principale de l'écorce pour améliorer l'idée de base de la seconde méthode. Nous avons par exemple ajouté la gestion des quelques écorces ayant plus de deux niveaux, comme l'écorce de platane. Enfin, les détails qui ne sont pas présents dans la macro-texture sont ajoutés par une micro-texture calculée de la même manière que dans la seconde méthode.

La combinaison de ces trois méthodes, en privilégiant d'abord la première qui fonctionne dans la plupart des cas, permet d'obtenir une carte de hauteur correspondant visuellement assez bien au véritable relief du modèle et nous permet donc d'obtenir de bons résultats proches du photo-réalisme souhaités comme le montrent les résultats présentés au Chapitre 5.

2.3 Synthèse de texture d'écorce

Grâce aux méthodes présentées au paragraphe 2.2, nous parvenons à extraire, pour la majorité des espèces d'écorces, un relief visuellement cohérent par rapport à un modèle de texture d'écorce quelconque donné par l'utilisateur. Ce modèle est une photographie couleur rectangulaire à laquelle on associe donc une carte de hauteurs en niveaux de gris de mêmes dimensions pour former un échantillon de texture en relief ($2D\frac{1}{2}$).

Or, il suffit d'observer la complexité des branches d'un arbre pour comprendre que l'application de cette texture rectangulaire n'est pas aussi triviale que de coller du papier peint sur un mur. Il se pose en effet plusieurs problèmes. Tout d'abord, il y a un problème de discontinuité dû principalement à la forme des embranchements où la texture se sépare

pour suivre chaque branche. Toutefois, dans ce cas précis, la discontinuité n'est en fait pas vraiment un problème car elle est justement présente dans la nature comme on peut le voir par exemple sur les photographies de la Figure 25 au paragraphe 1.3.3.5. Le vrai problème de discontinuité qui se pose est dû au modèle photographique de texture qui ne « boucle » pas, c'est-à-dire qu'il ne peut pas être juxtaposé plusieurs fois à lui-même sans qu'on voie des discontinuités sur les bords (parfois appelés « coutures ») de la texture. Enfin, le dernier problème est que, quand bien même on modifierait cette texture pour qu'elle boucle sans discontinuités, on remarquerait un effet de répétition périodique du modèle qui ne paraîtrait pas du tout naturel.

La solution que nous avons retenue est donc de synthétiser une texture aperiodique sur un cylindre « infini », afin de supprimer les discontinuités aux bords du modèle ainsi que l'effet visible de répétition périodique d'un même motif, en associant une couleur et une élévation à chaque coordonnée cylindrique (u,v) telles que u donne une rotation autour de l'axe et v une distance par rapport à la base de l'axe. Le cylindre est « infini » en longueur pour pouvoir gérer n'importe quelle longueur de branches et boucle en largeur pour envelopper les branches qui sont de forme relativement cylindrique. Il restera donc uniquement le « non-problème » de la discontinuité naturelle qu'il *faut* conserver à l'intérieur des angles formés entre les branches « filles » d'un embranchement. La façon dont cette texture est appliquée sur les branches est détaillée au Chapitre 3.

Nous nous sommes cependant intéressés à l'économie de mémoire. En effet, dans le but d'obtenir des résultats de qualité, nous avons besoin de textures d'écorce d'assez haute résolution et il faut donc limiter la place qu'elles occupent en mémoire. La notion d'infini que nous recherchons, même si nous la restreignons à de très larges textures de dimensions finies, serait donc trop lourde à stocker, ce qui implique donc que nous ne pouvons pas nous permettre de synthétiser ce cylindre en entier et le stocker en mémoire.

Il faudrait donc, a priori, trouver un moyen de calculer la texture « à la volée », ce qui pose alors un problème de rapidité de calcul. Certaines méthodes comme l'échantillonnage à base de morceaux de texture (« *patch-based sampling* ») [136] [70] sont très rapides et efficaces mais pas suffisamment rapides pour synthétiser la texture au cours du rendu par lancer de rayons que nous souhaitons faire pour visualiser les branches par surfaces implicites perturbées car cela nécessite de calculer plusieurs millions de fois la

couleur et la hauteur d'un point de l'espace (cf. Chapitre 3). Il nous faut trouver une technique qui permette d'obtenir encore plus rapidement le couple (*couleur, hauteur*) correspondant aux coordonnées (u, v) sur le cylindre infini. D'autre part, pré-calculer une très large texture qui boucle permettrait d'obtenir rapidement ces informations durant le rendu mais consommerait trop de mémoire dans le cas d'une texture d'écorce en haute résolution.

Les problèmes de mémoire et de temps de calcul étant souvent opposés mais tous deux très importants, nous avons choisi une méthode intermédiaire qui s'avère très proche de la méthode proposée très récemment par Cohen et al. [24] qui se sont basés sur une idée de Stam [118] pour utiliser des « *Wang Tiles* » [128] [129] (« tuiles de Wang »), dont le but consiste à assembler de façon stochastique un nombre réduit d'échantillons différents ayant la particularité de pouvoir être juxtaposés côte à côte sans discontinuités visibles. Cependant, contrairement aux « *Wang Tiles* », nous avons choisi de n'utiliser que des échantillons ayant tous les mêmes bords car nous avons constaté que, sur des branches, les bords des échantillons ne représentent qu'une faible surface dont la répétition est imperceptible et le fait que tous les bords soient identiques permet d'avoir moins d'échantillons à stocker tout en offrant un plus grand nombre de possibilités d'arrangements de ces échantillons les uns à côté des autres. Cela permet donc d'utiliser assez peu de mémoire tout en pouvant accéder à la texture très rapidement au cours du rendu puisque la synthèse des échantillons se fait lors d'un pré-calcul.

La méthode que nous proposons, pour synthétiser chaque échantillon, est décrite au paragraphe 2.3.1. Elle fait en sorte que les bords de tous les échantillons soient les mêmes en ne faisant varier que l'intérieur de chaque échantillon, ce qui est suffisant pour que l'on ne remarque pas d'effet de répétition. Nous présentons ensuite, au paragraphe 2.3.2, la méthode qui indexe chaque échantillon sur le cylindre et qui renvoie une couleur et une élévation pour n'importe quelle coordonnée (u, v) . Enfin, nous illustrons quelques résultats de synthèse d'écorces sur un plan au paragraphe 2.3.3.

2.3.1 Synthèse d'un ensemble d'échantillons de texture juxtaposables

Nous avons donc choisi de synthétiser un petit ensemble de N_E échantillons de texture d'écorce, juxtaposables sans discontinuités, ayant tous les mêmes dimensions

(L_E, H_E) , et de les distribuer de façon stochastique sur l'arbre (en fait, sur chaque branche rapportée à un cylindre comme nous le verrons en détail au paragraphe 3.3.1). Un ensemble d'environ huit échantillons de dimensions supérieures ou égales au modèle suffit pour donner des résultats très satisfaisants. Dans ce chapitre nous illustrons la partie colorée de la texture mais la carte de hauteur associée doit, bien sûr, subir exactement les mêmes traitements.

La continuité d'une bordure d'un échantillon de texture rectangulaire à la bordure opposée de n'importe quel échantillon n'est garantie que si tous ces échantillons ont la même bordure. C'est une contrainte assez forte car elle rend l'effet de répétition plus difficile à cacher. Malgré leurs bords communs, tous ces échantillons doivent donc paraître suffisamment différents tout en restant représentatifs du modèle original.

Il existe de nombreuses méthodes de synthèse de textures mais seulement certaines d'entre elles permettent de synthétiser des textures qui bouclent. La plupart des méthodes basées sur une structure pyramidale [7] [17] permettent de synthétiser ce genre de textures; à condition que le modèle boucle, ce qui n'est pas le cas des textures naturelles comme l'écorce. D'autres méthodes comme celles d'Ashikmin [4], de Drori et al. [34], d'Efros et Leung [35], d'Efros et Freeman [36], de Hertzmann et al. [54] et de Wei et Levoy [133] peuvent extrapoler une texture en y bouchant des trous et elles permettraient donc de synthétiser l'intérieur de la texture après avoir synthétisé les bordures mais elles ne permettent pas a priori de synthétiser les bords de façon à ce qu'ils bouclent.

Une méthode efficace qui permet de garantir le bouclage de la texture est l'échantillonnage à base de morceaux de texture (« *patch-based sampling* ») [136] [70] grâce auquel on peut coller des morceaux de textures qui débordent d'un côté de la texture pour boucler de l'autre côté et qui seraient pris à l'intérieur du modèle et donc continus. C'est cette méthode que nous utilisons.

La première étape de la synthèse des échantillons est donc de générer les bordures d'un échantillon de base que nous appelons E_0 . Tous les échantillons synthétisés E_i (i allant de 1 à N_E) seront basés sur E_0 en ne remplissant que l'intérieur avec des morceaux de textures issus du modèle comme nous l'expliquons ci-dessous (voir paragraphe 2.3.1.1). Les méthodes de « *patch-based sampling* » [136] [70] sur lesquelles nous nous basons

présentent néanmoins deux légers problèmes qui sont l'apparition de zones floues aux bords des morceaux dupliqués et une possible décoloration de la texture lors du mélange de zones contrastées. Nous expliquons ces deux problèmes de flou et de décoloration plus en détail, respectivement aux paragraphes 2.3.1.2 et 2.3.1.3, en proposant une façon de les corriger. Avec nos améliorations, cette méthode très simple donne des résultats très satisfaisants pour presque tous les types d'écorces (ainsi que pour beaucoup d'autres textures « chaotiques »).

2.3.1.1 Copie de morceaux de texture

Un constat évident est que la texture du modèle est discontinue sur ses bords mais continue à l'intérieur. Par conséquent, l'idée de base est de coller, sur les bords de E_0 , des morceaux continus extraits de façon stochastique à partir du modèle, comme dans la méthode de Xu et al. [70] reprise par Liang et al. [136]. La technique consiste donc à dupliquer des morceaux rectangulaires sur la texture à synthétiser jusqu'à en recouvrir toute la surface. Afin de ne pas générer de discontinuités sur les bords des morceaux dupliqués, ceux-ci sont mélangés progressivement en faisant varier l'opacité de chaque pixel par rapport à sa distance au bord du morceau.

Il est tout d'abord important de choisir des dimensions appropriées pour ces morceaux. S'ils sont trop larges, la répétition des motifs risque de se remarquer, tandis que s'ils sont trop petits, la macro-texture sera perdue. Considérant que le modèle devrait avoir une largeur $L_{modèle}$ (plus petite que sa hauteur $H_{modèle}$) représentant entre un quart et un huitième du périmètre de la branche où du tronc d'où il provient et, considérant que l'échelle des détails des différentes espèces d'écorces par rapport à la largeur du modèle est plutôt constante, nous avons remarqué par expérience que des morceaux à peu près carrés, un peu plus larges qu'un quart de la largeur du modèle, offrent de bons résultats. Le nombre de morceaux de texture requis pour couvrir chaque échantillon en largeur devrait donc être : $N_{morceauxL} = 4$ dans le cas où l'on a choisi de donner aux échantillons la même taille qu'au modèle. Sinon, il faudra augmenter cette valeur en proportion. Le nombre $N_{morceauxH}$ de morceaux en hauteur est alors donné par l'équation (20) :

$$(20) \quad N_{morceauxH} = \lfloor (N_{morceauxL} \times H_E / L_E) + 0,5 \rfloor$$

Les dimensions ($L_{\text{morceau}}, H_{\text{morceau}}$) d'un morceau de texture, définies par les équations (21) et (22), sont composées de trois parties qui représentent respectivement la taille minimale de la partie opaque du morceau, une valeur aléatoire pour que tous les morceaux n'aient pas exactement les mêmes dimensions et deux fois la taille de la zone de mélange (une de chaque côté de la partie opaque).

$$(21) \quad L_{\text{morceau}} = L_{\text{opaqueMin}} + \text{hasard}\left(\frac{L_{\text{opaqueMin}}}{2}\right) + 2 \times L_{\text{mélange}}$$

$$(22) \quad H_{\text{morceau}} = H_{\text{opaqueMin}} + \text{hasard}\left(\frac{H_{\text{opaqueMin}}}{2}\right) + 2 \times H_{\text{mélange}}$$

La fonction $\text{hasard}(x)$ retourne une valeur réelle aléatoire dans l'intervalle $[0, x[$ et ($L_{\text{mélange}}, H_{\text{mélange}}$) sont les dimensions des zones de mélange décrites plus bas (cf. paragraphe 2.3.1.2). Les dimensions minimales ($L_{\text{opaqueMin}}, H_{\text{opaqueMin}}$) de la partie opaque de chaque morceau de texture sont données par les équations (23) et (24) et sont supérieures ou égales aux dimensions requises pour recouvrir l'échantillon afin d'éviter qu'il reste des trous dans les textures synthétisées.

$$(23) \quad L_{\text{opaqueMin}} = \left\lceil \frac{L_E}{N_{\text{morceau}L}} \right\rceil$$

$$(24) \quad H_{\text{opaqueMin}} = \left\lceil \frac{H_E}{N_{\text{morceau}H}} \right\rceil$$

L'ordre dans lequel les morceaux seront dupliqués n'est pas très important mais il n'est pas recommandé de les coller séquentiellement. Nous avons donc choisi de faire ce traitement en deux passes, ce qui pourrait être comparé à coller les morceaux d'abord sur les cases noires d'un damier puis sur les cases blanches. La Figure 57 illustre le processus complet de génération de E_0 et montre que seule la première colonne et la première ligne sont dupliqués dans E_0 puisque les bords opposés sont traités en même temps par débordement.

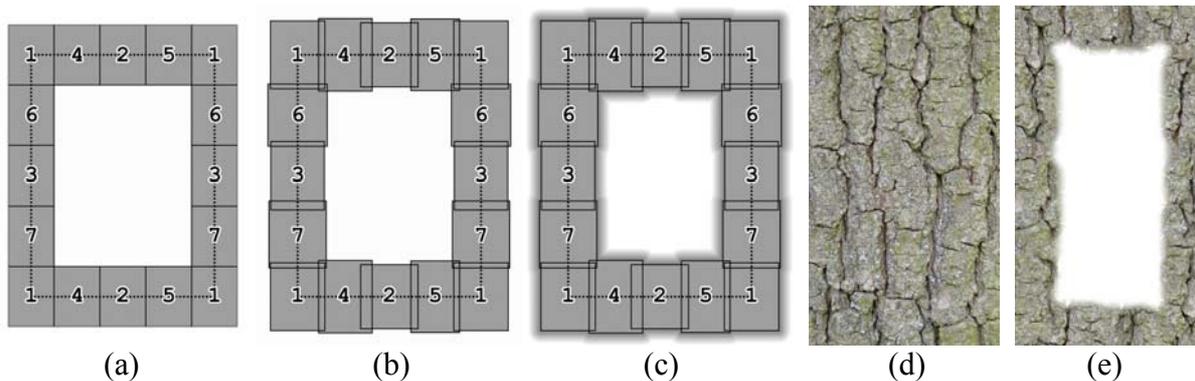


Figure 57 : Création d'un échantillon de base E_0 qui boucle. Les pointillés sur les images (a) à (c) représentent les dimensions des échantillons et les rectangles continus représentent la partie opaque des morceaux de texture dupliqués. a) Collage de morceaux de même taille sans mélange (les morceaux sont numérotés dans l'ordre où ils sont collés) ; b) ajout d'une valeur aléatoire à la taille ; c) rajout des zones de mélange (solution retenue) ; d) exemple de modèle ; e) l'échantillon de base E_0 correspondant au modèle et qui « boucle ».

Ensuite, pour synthétiser les N_E échantillons différents E_i à partir de E_0 , nous commençons par copier E_0 dans tous les E_i puis, comme pour la génération de E_0 , nous prenons aléatoirement des morceaux de texture dans le modèle, et nous les dupliquons à l'intérieur de chaque échantillon E_i afin qu'ils aient tous l'air différents tout en conservant les mêmes bords (ceux de E_0). La Figure 58 illustre quatre échantillons différents synthétisés à partir de l'écorce utilisée dans la Figure 57.

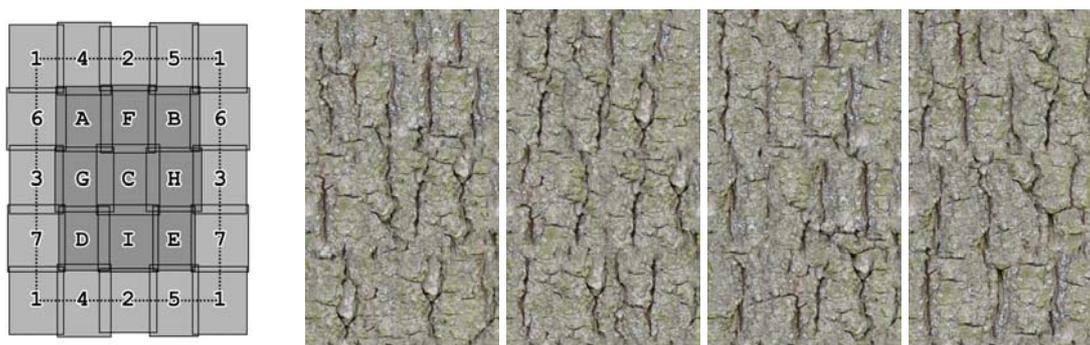


Figure 58 : Duplication des morceaux de texture (représentés par des lettres dans l'ordre où ils sont collés sur l'image de gauche) au centre de chaque échantillon E_i initialisés avec E_0 pour synthétiser différents échantillons d'une même texture. Les quatre échantillons synthétisés (à droite) sont visuellement différents mais leurs bordures sont identiques.

Il faut cependant respecter la contrainte imposant que les morceaux dupliqués dans la partie centrale des échantillons ne doivent en aucun cas déborder de l'échantillon afin de conserver la propriété de juxtaposition sans discontinuité des échantillons.

Grâce à la partie aléatoire (que nous appellerons *alea*) dans les équations (21) et (22) définissant les dimensions des morceaux de texture, la surface opaque de ces derniers est strictement plus large que la taille nécessaire pour éviter de laisser des trous dans la texture. Par conséquent, il n'est pas nécessaire de centrer précisément les morceaux de texture sur leur position théorique. Un décalage de cette position par une autre valeur aléatoire dans l'intervalle $[-alea/2 ; +alea/2]$ permet d'atténuer légèrement l'effet – déjà très limité – de bloc pouvant rendre parfois visible la forme rectangulaire des morceaux de texture dans le cas de textures aux motifs très larges.

2.3.1.2 Eviter les discontinuités sans ajouter d'effet de flou

La technique décrite ci-dessus est à ce stade très proche du « *patch-based sampling* » [136] [70] et présente un problème qui peut être négligeable pour certaines écorces mais qui est en général visible : des zones « floues » (de fondu entre deux morceaux de texture distincts) apparaissent dans les textures synthétisées aux endroits où l'on a mélangé des morceaux contrastés, mais ce mélange est nécessaire pour éviter les discontinuités aux bords des morceaux de texture dupliqués. De plus, ces contours rectangulaires deviennent plus visibles lorsque le contraste est important. Par conséquent, nous avons quelque peu modifié cette méthode pour que le collage des morceaux soit fait avec un mélange progressif spécial qui tente de réduire l'impression de flou dans les zones de mélange, ainsi que l'aspect rectangulaire des morceaux collés.

Réduire la taille de la zone de mélange par interpolation accentue la visibilité de la forme rectangulaire des morceaux dupliqués tandis que l'élargir augmente l'effet de flou sans vraiment dissimuler la forme rectangulaire des morceaux de texture car l'œil humain arrive à distinguer assez facilement des zones nettes (parties opaques de chaque morceau de texture) entourées par des zones floues (zones de mélange). L'utilisation de morceaux non rectangulaires comme avec la méthode des « *lapped textures* » [96] ou les « particules de texture » [33] est un très bon moyen d'éliminer le problème de la forme rectangulaire des morceaux et permet ainsi d'utiliser une surface de mélange beaucoup plus réduite

autour de la zone opaque afin de ne plus poser de problème de flou. Le seul problème est que cette méthode nécessite une interaction avec l'utilisateur qui doit définir manuellement la forme des morceaux.

Cette idée est néanmoins efficace et il est donc intéressant de trouver un moyen de modifier automatiquement le contour des morceaux à coller. Notre but étant de réduire la surface de mélange tout en réduisant l'aspect rectangulaire des morceaux de texture extraits du modèle de façon automatique, nous proposons une méthode permettant de modifier l'opacité du contour de chaque morceau de texture à dupliquer (rectangulaire, à la base) en fonction des éléments de texture du morceau et de l'endroit où il sera appliqué dans l'échantillon E_i . Dans la méthode de Xu et al. [136], l'opacité des bords varie linéairement de 0% à 100%, des bords vers la partie opaque rectangulaire du morceau, suivant la distance $dist_{opaque}$ du pixel p traité par rapport à la zone opaque. Cette distance, rapportée aux dimensions $(L_{mélange}, H_{mélange})$ de la zone de mélange est donnée par l'équation (25) utilisant les équations (26) et (27).

$$(25) \quad dist_{opaque}(p) = \sqrt{\left(\frac{dist_x(p)}{L_{mélange}}\right)^2 + \left(\frac{dist_y(p)}{H_{mélange}}\right)^2}$$

$$(26) \quad dist_x(p) = \max(0, L_{mélange} - x_p, L_{mélange} + x_p + 1 - L_{morceau})$$

$$(27) \quad dist_y(p) = \max(0, H_{mélange} - y_p, H_{mélange} + y_p + 1 - H_{morceau})$$

Nous proposons d'introduire un second paramètre : la distance de couleur $dist_{couleur}$ entre les deux pixels à mélanger, correspondant à p respectivement dans l'échantillon E_i et dans le morceau de texture à coller. En effet, l'effet de flou étant plus visible lorsque des pixels contrastés sont mélangés, on peut le réduire considérablement en modifiant l'opacité en fonction des variations de contraste.

Le calcul de $dist_{couleur}(x,y)$ peut être fait en convertissant les deux couleurs RVB $x(R_x, V_x, B_x)$ et $y(R_y, V_y, B_y)$ à comparer dans l'espace $L^*a^*b^*$, créé pour offrir une métrique linéaire perceptuelle par rapport à l'œil humain, et de calculer la distance Euclidienne dans ce nouvel espace. Mais nous avons utilisé un calcul plus simple, donné par l'équation (28), qui se base surtout sur la luminosité des couleurs au moyen des coefficients de la

recommandation CIE Rec.709 donnant la luminosité (« *luma* ») des teintes pures RVB. Cette formule n'est pas très juste car elle supposerait que l'œil humain perçoit les différences de luminosité de façon linéaire dans l'espace RVB, ce qui n'est pas vrai ; mais elle donne des résultats très satisfaisants pour l'application que nous en faisons et le choix d'utiliser des coefficients dédiés à la luminosité (appelée Y_{709}) peut se justifier par le fait que la principale information contenue dans une couleur est sa luminosité. De plus, le processus d'applications de morceaux d'écorce chaotiques étant aléatoire, il n'est pas absolument nécessaire d'employer une distance de couleurs extrêmement précise. La distance $dist_{couleur}$ varie donc de 0 (couleurs identiques) à 1 (comparaison du noir avec du blanc).

$$(28) \quad dist_{couleur}(x, y) = 0.212671|R_x - R_y| + 0.715160|V_x - V_y| + 0.072169|B_x - B_y|$$

En fait, l'idée de base est de réduire la taille de la « zone effective de mélange » (ZEM) pour atténuer l'effet de flou, mais d'utiliser une « zone théorique de mélange » (ZTM) assez large dans laquelle « flotte » la ZEM afin de réduire l'aspect rectangulaire du morceau de texture. La valeur $taille_{mélange}$ représente la proportion de la ZEM par rapport à la ZTM. Si $taille_{mélange} = 0$, alors il n'y a pas de mélange et la transition est abrupte entre la surface opaque du morceau de texture et le reste de l'échantillon synthétisé. A l'inverse, si $taille_{mélange} = 1$, la ZEM est égale à la ZTM. Nous avons constaté que 0,5 est une bonne valeur de $taille_{mélange}$.

Notons que les dimensions ($L_{mélange}, H_{mélange}$) de la ZTM dépendent de l'orientation de l'écorce définissant le rapport $ratio_{texture}$ donné par l'équation (15). Par exemple, la texture « horizontale » illustrée dans la Figure 55 nécessite une zone de mélange plus large que haute. Ces dimensions sont données par les équations (29) et (30) :

$$(29) \quad L_{mélange} = \sqrt{\frac{L_{opaqueMin}^2 + L_{opaqueMin}^2}{1 + (1/ratio_{texture})}}$$

$$(30) \quad H_{mélange} = \frac{L_{mélange}}{ratio_{texture}}$$

Il reste à définir la façon dont la ZEM doit être utilisée à l'intérieur de la ZTM ; et la réponse est dans la distance de couleurs $dist_{couleur}$. Nous plaçons la ZEM à proximité du bord intérieur (proche de la zone opaque) de la ZTM lorsque les pixels à mélanger sont très contrastés ($dist_{couleur}$ proche de 1) et nous plaçons la ZEM à proximité du bord extérieur lorsque les pixels à mélanger sont assez similaires ($dist_{couleur}$ proche de 0) ce qui permet de faire varier le contour du morceau en fonction de la texture. La forme de la ZEM est calculée à l'aide d'un cosinus (entre 0 et π) pour former une sigmoïde, produisant ainsi une transition douce entre les trois différentes zones (voir partie gauche de la Figure 59) : partie complètement opaque, ZEM, partie complètement transparente.

Pour chaque morceau de texture, nous calculons par conséquent une sorte de masque de mélange (voir Figure 60) qui donne l'opacité de chaque pixel p du morceau. L'équation (31), faisant appel aux équations (32) et (33) permet de calculer cette opacité :

$$(31) \quad \text{opacité}(p) = \begin{cases} 1 & \text{si } dist_{opaque} \leq déb_{mél} \\ \left(1 + \cos \frac{\pi \times (dist_{opaque} - déb_{mél})}{taille_{mélange}}\right) / 2 & \text{si } déb_{mél} < dist_{opaque} < fin_{mél} \\ 0 & \text{si } dist_{opaque} \geq fin_{mél} \end{cases}$$

$$(32) \quad déb_{mél} = dist_{couleur} \times (1 - taille_{mélange})$$

$$(33) \quad fin_{mél} = déb_{mél} + taille_{mélange}$$

La valeur à partir de laquelle le pixel p du morceau de texture commence à être mélangé avec l'échantillon à synthétiser est représentée par $début_{mél}$. Une valeur inférieure à $début_{mél}$ signifie que le masque pour ce pixel est opaque (égal à 1). A l'inverse, $fin_{mél}$ représente la limite à partir de laquelle le masque de mélange de p est transparent (égal à 0). Comme le montre la Figure 59, une fois que la ZEM est positionnée en fonction de $dist_{couleur}$, pour chaque pixel p , nous trouvons l'opacité du mélange grâce à la distance $dist_{opaque}$ de p par rapport à la partie opaque du morceau.

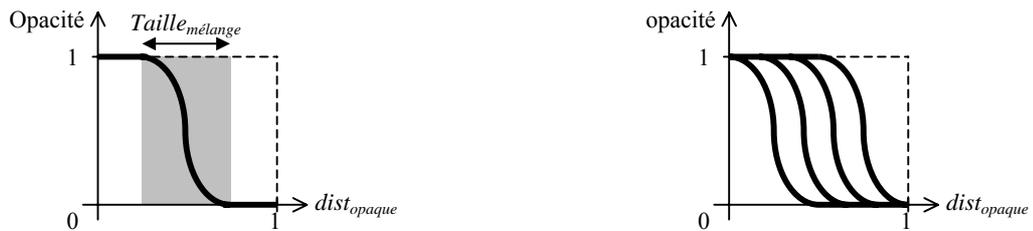


Figure 59 : Opacité de masque de mélange selon $dist_{opaque}$ dans la ZTM. A gauche : la zone grisée représente la ZEM. A droite : quatre exemples de positionnement de la ZEM dans la ZTM pour différentes valeurs de $dist_{couleur}$ (les courbes les plus à gauche représentent le mélange de pixels contrastés).

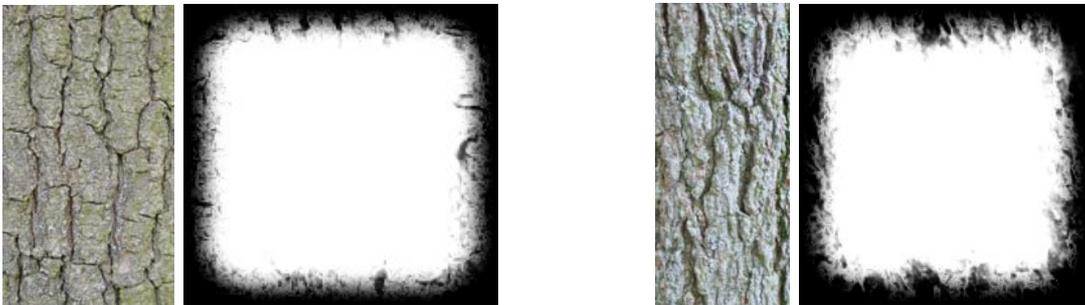


Figure 60 : Deux exemples de masques de mélange quelconques qui ont servi lors de la synthèse d'échantillons à partir du modèle d'écorce. Note: Dans cette figure, les masques ont été grossis par rapport aux modèles mais leur largeur est normalement légèrement supérieure à un quart de la largeur du modèle. Chaque pixel du masque dépend du pixel du morceau de texture à coller et du pixel de l'échantillon avec lequel il sera mélangé.

2.3.1.3 Eviter la perte de couleur

Le mélange des pixels les plus clairs de la texture (généralement blancs) avec les plus sombres (généralement noirs) donne un résultat grisâtre. Or, pour des textures plutôt monochromes ne contenant pas de tons gris, les pixels gris résultant du mélange peuvent devenir visibles et donc gênants. Il est alors nécessaire de coloriser les pixels gris résultant du mélange de pixels contrastés au moyen de la teinte moyenne du modèle.

Le calcul de cette teinte moyenne dépend encore une fois, comme pour le calcul de distance de couleurs, de l'espace couleur dans lequel on va travailler. La plupart des

espaces couleur (sur lesquels on peut trouver de nombreuses informations sur le site Internet de Charles Poynton [145]) séparant la luminosité de la teinte (ou chromaticité, souvent représentée à l'aide de deux composantes), tels que les espaces XYZ, L*a*b* et YUV, conviennent. Le traitement consiste donc à calculer la couleur RVB moyenne du modèle, puis à la transformer dans un des espaces couleur cités ci-dessus qui séparent la luminosité de la teinte (qui est la teinte moyenne recherchée).

Cependant, cette moyenne des pixels RVB doit être pondérée par l'importance de chaque pixel. En effet, les pixels noirs et blancs ne contiennent pas d'information de couleur et devraient donc être ignorés tandis que les pixels d'intensité moyennes sont supposés plus significatifs. L'intensité Y_{709} des pixels est calculée en faisant la somme des composantes RGB pondérées chacune par les coefficients utilisés dans l'équation (28). La fonction *importance* définie par l'équation (34) reçoit en paramètre la luminosité d'un pixel de couleur et renvoie 1 pour les pixels de luminosité moyenne, 0 pour des pixels noirs ou blancs et des valeurs intermédiaires pour les niveaux de gris intermédiaires.

$$(34) \quad \text{importance}(x) = 1 - 2 \times |x - 0.5|$$

Il reste à déterminer la proportion de la teinte moyenne qui sera appliquée à la couleur c qui résulte du mélange des deux pixels (du morceau et de l'échantillon). Cette proportion dépend de l'opacité du masque de mélange et du contraste ($dist_{couleur}$) des deux pixels mélangés. Si l'opacité est proche de 50%, le mélange sera maximal. Cela signifie que la teinte réelle est plus susceptible d'avoir été altérée dans ce cas ; ce qui implique que la correction de teinte devrait être plus forte. Nous réutilisons donc notre fonction *importance* dans l'équation (35) pour calculer la proportion de la teinte moyenne $prop_{teinteMoy}$ à appliquer à la couleur c en fonction de l'opacité du masque de mélange:

$$(35) \quad prop_{teinteMoy} = dist_{couleur} \times \text{importance}(opacité)$$

Une fois que cette proportion est calculée, la luminosité de c est combinée avec la teinte moyenne et la couleur résultante *teinte* est mélangée avec c par la formule (36) pour obtenir la couleur corrigée $c_{corrigée}$:

$$(36) \quad c_{corrigée} = c \times (1 - prop_{teinteMoy}) + teinte \times prop_{teinteMoy}$$

2.3.2 Génération d'une texture infinie cylindrique

Comme nous le détaillons au paragraphe 3.3.1, nous avons besoin d'une fonction f_{tex} qui renvoie une couleur et une hauteur pour n'importe quel point de l'espace par rapport aux branches. La texture pour chacun de ces points est définie par coordonnées cylindriques (u,v) qui sont parfaitement adaptées à nos textures rectangulaires juxtaposables. Afin de rendre invisible l'effet de répétition de ces échantillons, nous proposons une méthode pour les arranger aléatoirement sur un cylindre infini qui servira de base pour texturer les branches.

Nous calculons une petite table ($table_E$) en deux dimensions, contenant des indices i représentant chacun des échantillons E_i choisis de façon stochastique et aperiodique. Il y a des contraintes à respecter dans le choix de ces indices afin de ne pas juxtaposer deux fois le même échantillon, ce qui est également valable aux bords de la table puisque celle-ci doit boucler en largeur, bien sur, puisqu'elle enveloppe un cylindre, mais aussi en hauteur car elle ne peut pas être vraiment infinie (voir Figure 61). La largeur L_{table} de la table est le nombre d'échantillons nécessaires pour faire le tour des branches de l'arbre (par exemple 4 échantillons). La hauteur de la table est la période de répétition des échantillons le long des branches. Considérant qu'une hauteur de 256 échantillons pour une largeur de 4 échantillons ne nécessiterait qu'un Kilo-octet en mémoire ($256 \times 4 = 1024$), il n'y a aucune raison de limiter ce nombre à de petites valeurs et on peut donc considérer notre table comme pratiquement « infinie », du moins supérieure à la hauteur nécessaire pour texturer les branches sur toute leur longueur.

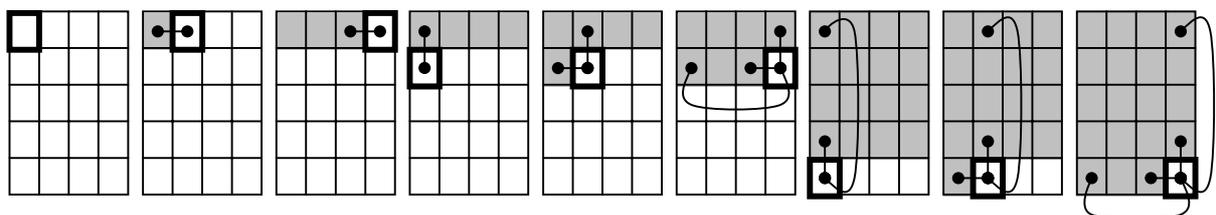


Figure 61 : Différentes étapes du processus de remplissage Markovien (linéaire) de la table contenant les indices des échantillons de textures synthétisés. La case en cours de traitement (en gras) doit recevoir un indice différent de ceux déjà assignés (cases grisées) à ses voisins (les liens indiquent les voisins à prendre en compte).

Une fois la table remplie, il ne reste donc plus qu'à définir la fonction f_{tex} qui retourne une couleur et une profondeur pour n'importe quelles coordonnées (u,v) . La première étape est de déterminer quel est l'indice i de l'échantillon E_i qui correspond aux coordonnées (u,v) grâce à l'équation (37) :

$$(37) \quad i(u,v) = table_E \left[\left(\frac{u}{L_E} \right) \bmod L_{table}, \left(\frac{v}{H_E} \right) \bmod H_{table} \right]$$

On n'a plus ensuite qu'à transformer les coordonnées (u,v) dans E_i . D'où la fonction $f_{tex}(u,v)$ définie par l'équation (38) :

$$(38) \quad f_{tex}(u,v) = E_{i(u,v)}(u \bmod L_E, v \bmod H_E)$$

2.3.3 Exemples d'écorces synthétisées

La Figure 62 montre quatre exemples de textures d'écorce (couleur et relief) dont nous avons synthétisé quatre échantillons juxtaposés en carré (2×2).

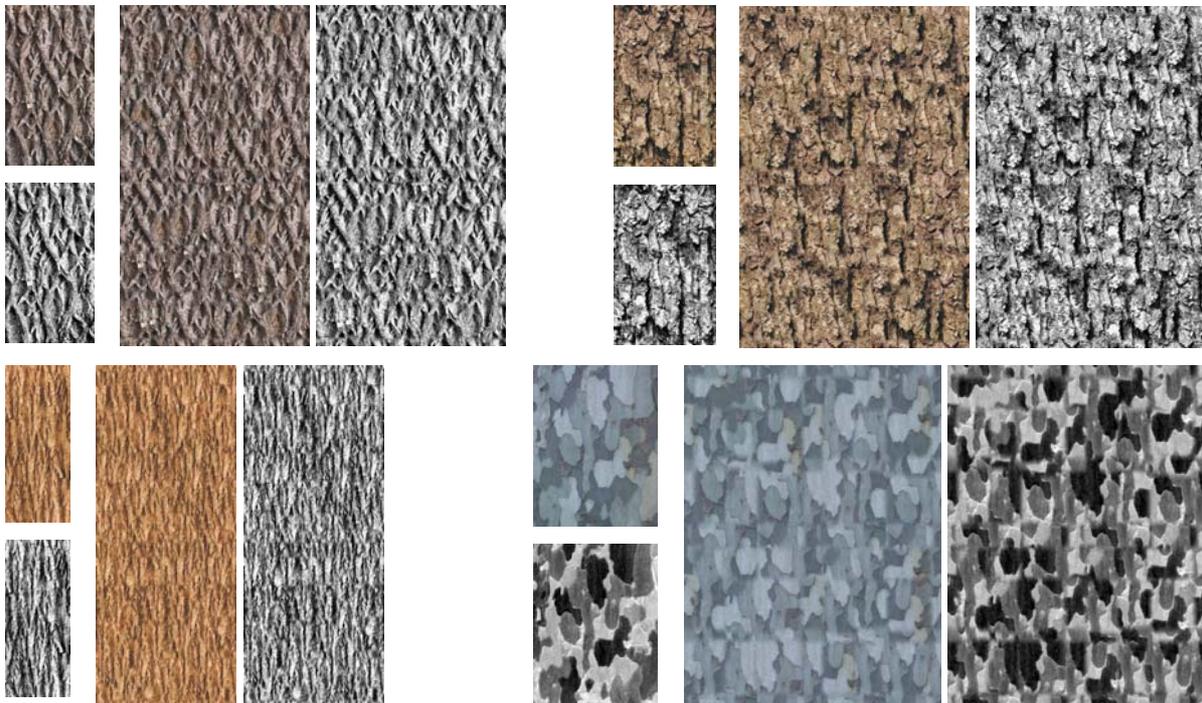


Figure 62 : Quatre écorces différentes synthétisées à partir de leurs modèles (cartes de couleurs et de hauteurs).

L'exemple en bas à droite de cette figure montre que la méthode n'est pas très efficace avec des textures comportant de larges zones plates bien distinctes car l'effet de flou ne peut pas être évité et on remarque la coupure des éléments de texture suivant une forme plutôt rectangulaire. Mais, les résultats restent cependant acceptables.

2.4 Conclusion

Dans le but final d'appliquer une texture rugueuse d'écorce sur les branches d'un arbre, nous avons cherché quelle méthode serait la plus simple pour l'utilisateur d'obtenir cette texture. En considérant d'un côté les méthodes de synthèse pures et de l'autre les méthodes « basées image », nous en avons conclu que ces dernières partaient avec un net avantage quant au réalisme visuel, mais que leur principal inconvénient est que les photographies sont en deux dimensions et ne contiennent donc pas explicitement d'information sur leur relief. D'autre part, les méthodes existantes, permettant d'extraire une carte de hauteurs à partir de photographies d'objets réels, nécessitent le plus souvent de prendre plusieurs photographies d'un même objet dans des conditions d'éclairage précises et requièrent parfois une installation matérielle encombrante, voire onéreuse. Ce domaine, englobe le « *shape from shading* » dont certains algorithmes peuvent traiter des images uniques (par oppositions aux images stéréoscopiques ou encore plus nombreuses, moins pratiques pour l'utilisateur) mais ces techniques sont peu efficaces avec des textures assez bruitées et/ou colorées comme l'écorce.

Dans un souci de rendre l'acquisition de textures d'écorces en relief accessible à tous, nous avons donc proposé un ensemble de méthodes permettant d'obtenir un échantillon d'écorce en relief à partir d'une simple photographie servant de modèle. Ces méthodes demandent, pour la plupart, très peu d'interaction avec l'utilisateur et permettent de retrouver un relief qui, bien qu'il ne soit pas le relief exact, reste cohérent visuellement avec le modèle en couleur.

Nous avons ensuite proposé une méthode pour synthétiser une texture infinie et visuellement non répétitive d'écorce à partir de l'échantillon dont nous avons extrait le relief, en vue de pouvoir l'appliquer par la suite sans répétition ni discontinuités sur un arbre comme nous le proposons au chapitre suivant.

Chapitre 3

Application de texture aux branches

3.1 Introduction

La structure des arbres est très complexe et il n'est pas facile d'y appliquer une texture, qui plus est, en relief comme nous voulons le faire afin d'offrir un rendu de l'écorce le plus réaliste possible. De plus, nous souhaitons représenter les branches des arbres à l'aide de surfaces implicites mélangées afin d'adoucir leurs jonctions, ce qui complique encore a priori l'application de texture sur ces branches.

Nous avons choisi de représenter les branches à l'aide de cone-sphères [83] car c'est une représentation qui convient très bien à la représentation du squelette des arbres par segments (cf. paragraphe 1.3.3). Et, nous les représentons par surfaces implicites pour les trois raisons suivantes :

- Elles permettent d'adoucir les jonctions des branches en les mélangeant, ce qui augmente sensiblement le réalisme visuel.
- Elles permettent aussi de réduire considérablement l'utilisation de la mémoire (qui est très sollicitée avec des formes aussi complexes que des arbres très détaillés) en comparaison avec une représentation polygonale, ce qui peut se révéler particulièrement utile pour visualiser des forêts composées de plusieurs arbres différents.
- Enfin, elles peuvent faciliter le déplacement de surface (« *displacement mapping* »), d'une certaine proportion de la distance de la surface par rapport au squelette, pour représenter le relief de l'écorce. De plus, comme la paramétrisation de texture de notre méthode provient directement du squelette de l'arbre, elle est relativement facile à appliquer aux surfaces implicites puisqu'elles sont aussi basées sur le squelette.

Cependant, le choix d'utiliser des surfaces implicites présente un problème important : nous devons définir une méthode de mélange de surfaces qui évite le problème

couramment rencontré de gonflement aux jonctions des branches (voir Figure 63), tout en maintenant un contrôle total de la profondeur du relief de l'écorce 3D.

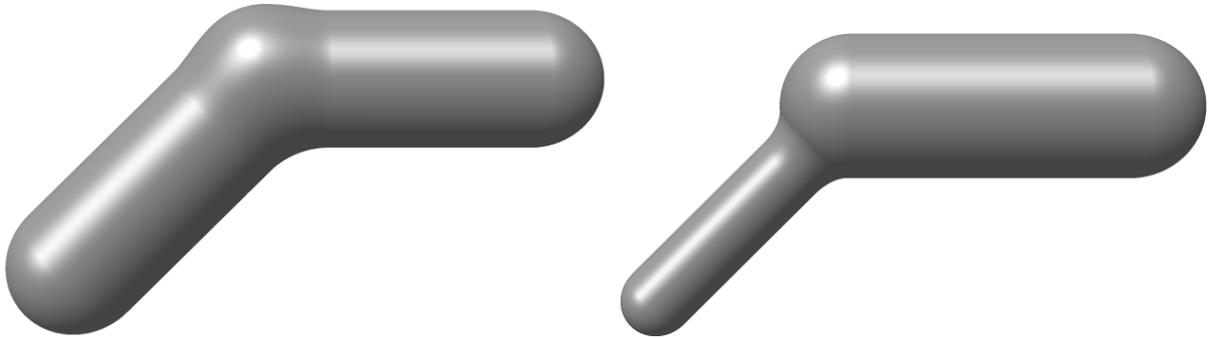


Figure 63 : A gauche : gonflement couramment rencontré à la jonction de deux objets implicites mélangés et de dimensions proches. A droite : si l'un des deux objets cylindriques a un champ de potentiel de rayon inférieur à celui de l'autre objet, alors il n'y a pas de gonflement.

En effet, la plupart des méthodes conçues pour éliminer ce gonflement [15] [16] [111] utilisent des fonctions de potentiels non linéaires qui rendent les modifications de profondeur linéaires difficiles à contrôler, ce qui ne posait pas de problème dans leur contexte d'origine puisque presque tous les travaux sur le mélange de surfaces implicites ont été faits pour des objets lisses. Il est vrai que, lorsqu'on souhaite adoucir la transition entre deux objets, on ne pense pas de prime abord à ce que ces objets soient rugueux. Toujours est-il que le problème de l'élimination du gonflement n'a toujours pas été complètement résolu.

Ferley et al. [37] ont proposé une méthode pour modéliser des embranchements de formes implicites mélangées ; mais leur méthode rajoute beaucoup de matière aux jonctions de branches, ce qui rend plus difficile l'application de textures, sans même parler du relief, notamment avec la méthode simple que nous proposons au paragraphe 3.3.

Les jonctions de branches adoucies font que la surface des arbres n'est a priori pas aussi triviale à texturer que dans le cas de simples formes cylindriques. Le thème du placage de texture sur des surfaces arbitraires a fait l'objet d'intenses recherches ces derniers temps [33] [86] [96] [116] [125] [138]. Cependant, la plupart de ces méthodes ont

été conçues pour des polyèdres (objets explicites dont la surface est formée d'un ensemble de triangles) et/ou ne conviennent pas à des textures 3D d'écorces et/ou produisent de larges informations de texture en termes de consommation de mémoire (soit à cause d'atlas de textures surdimensionnés, soit à cause de trop nombreuses coordonnées de positionnement de morceaux de textures, en raison de la surface totale très large que représentent les branches) et sont donc toutes relativement difficiles à mettre en œuvre pour des arbres. De plus, la représentation par polygones des surfaces implicites mélangées ne s'adapte a priori pas facilement à des arbres finement détaillés, surtout dans le cas où ils seraient intégrés dans une forêt, en raison d'un nombre très important de polygones et de la grande consommation de mémoire des méthodes pour y appliquer une texture.

En fait, la paramétrisation de texture sur des surfaces implicites est un problème assez difficile dans le cas général. Zonenschein et al. [143] ont proposé une méthode qui projette des particules partant de la surface d'un objet implicite à texturer et qui se dirigent vers une surface simple de référence que l'on sait texturer et qui englobe cet objet. La couleur atteinte par la particule sert alors pour texturer le point de départ de la particule sur les objets complexes. Cette méthode fonctionne très bien dans les cas où l'on peut définir une surface servant de support à la texture, comme par exemple un cylindre autour d'un segment de branche. Elle permet également de gérer les coudes formés par deux segments de branche consécutifs. Cependant, outre une relative lenteur, cette technique donne des résultats moins bons lorsque plus de deux objets doivent être représentés car elle ne propose alors que le mélange des textures par interpolation. Alors qu'il serait plus réaliste de voir la texture se séparer et suivre chacune des directions d'un embranchement, cette méthode ne le permet donc pas.

Tigges et Wyvill [121] ont proposé une méthode un peu moins généraliste mais permettant d'appliquer une texture sur des objets implicites quelconques. Toutefois, la transition de la texture d'un objet à l'autre grâce à leur méthode se fait encore une fois systématiquement par mélange.

Par conséquent, nous présentons une méthode « directe » – et donc économe en mémoire – pour appliquer une texture 3D sur des branches d'arbres, par une paramétrisation assez simple des textures synthétisées au Chapitre 2, sur des formes cylindriques (basées sur le squelette de l'arbre) mélangées de façon implicite.

Ce chapitre est donc organisé de la manière suivante. Le paragraphe 3.2 définit sous forme implicite l'élément de base de nos branches qu'est le cône-sphère et propose une façon de le mélanger avec d'autres cône-sphères afin de représenter un enchaînement lisse des segments de branches à leurs jonctions et embranchements. Le paragraphe 3.3 explique comment appliquer une texture en relief par déplacement de surface sur ces branches. Le paragraphe 3.4 présente une méthode pour appliquer des éléments de texture qui n'auraient pas pu être représentés par la carte de hauteur des écorces que nous synthétisons. Enfin, nous concluons sur les atouts et les quelques inconvénients de notre représentation.

3.2 Surfaces implicites

Les surfaces implicites sont relativement bien adaptées à la structure des arbres qui est définie par un squelette (cf. paragraphe 1.3.2), notamment par des L-systèmes [73] [74] [84] [98] [99]. En effet, elles permettent de représenter implicitement une « isosurface » définie par une fonction de distance donnant un potentiel par rapport au squelette ; ce potentiel pouvant être mélangé à celui des autres segments de branches adjacents pour adoucir la jonction de ces derniers.

Au paragraphe 1.3.3, nous avons vu que les formes les mieux adaptées aux segments de branches d'arbres sont les cône-sphères et les cylindres généralisés, mais ces derniers nécessitent, pour un rendu par lancer de rayons, des calculs d'intersection trop complexes pour être représentés par des surfaces implicites avec déplacement de surface. Nous avons donc choisi de donner du volume aux segments de branches au moyen de cône-sphères dont nous donnons une définition implicite au paragraphe 3.2.1. Nous présentons ensuite, au paragraphe 3.2.2, une méthode pour adoucir la jonction de cône-sphères adjacents.

3.2.1 Choix du volume de base

Les cône-sphères ont été introduits par Max [83] pour représenter des structures arborescentes en les mettant bout à bout. Cependant, alors qu'il est très facile de définir un volume implicite cylindrique (voir Figure 64, à gauche) à partir d'un segment du squelette, il est plus difficile de représenter un cône-sphère de façon implicite à cause de la distance de la surface au squelette qui varie le long de ce dernier.

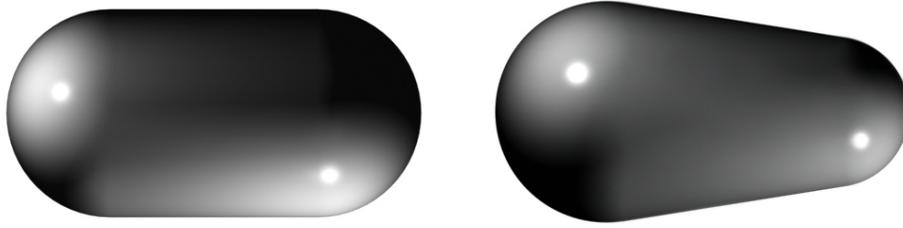


Figure 64 : A gauche : un volume implicite cylindrique. A droite : un cône-sphère.

Un cylindre implicite avec des hémisphères à ses extrémités, comme celui de la Figure 64, peut-être formulé par l'équation (39) :

$$(39) \quad f_{cyl}(P) = \frac{d(P,S)}{R} - 1 = 0$$

Dans l'équation ci-dessus, R représente le rayon du cylindre et $d(P,S)$ renvoie la distance Euclidienne du point P au segment S . La fonction f_{cyl} renvoie donc 0 pour les points situés sur la surface du cylindre, une valeur positive hors du cylindre et une valeur négative à l'intérieur (et -1 sur le squelette).

La formule est très similaire pour les cône-sphères, à la différence près que R varie entre les deux sphères formant les extrémités du cône-sphère, ce qui complique relativement le problème. Comme le montre la Figure 65 et contrairement à la représentation de Tigges et Wyvill [121], le rayon ne varie pas linéairement de R_1 à R_2 entre les deux extrémités du segment, mais entre les points de tangence du cône avec les sphères.

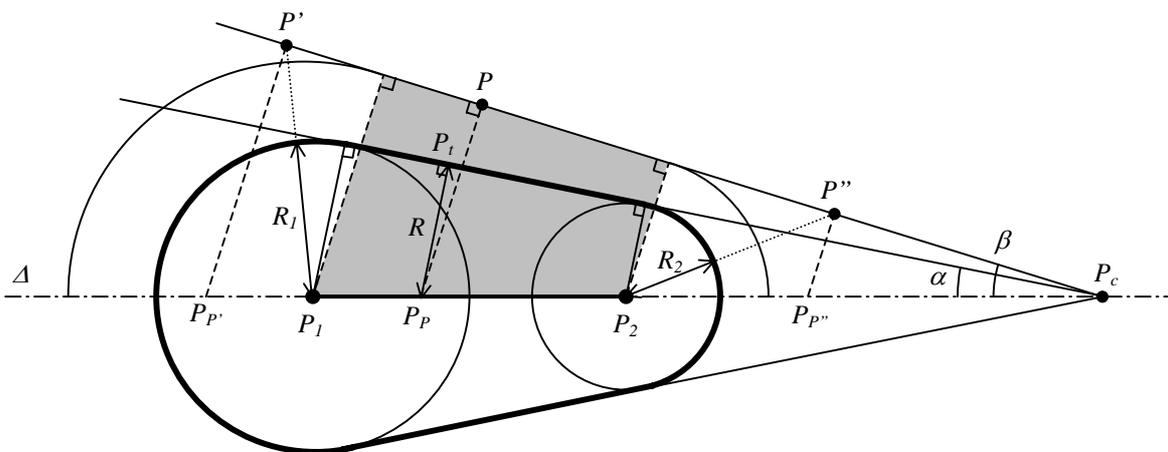


Figure 65 : Schéma d'un cône-sphère (contour en gras) en deux dimensions.

La fonction f définissant le potentiel d'un point P quelconque par rapport au cône-sphère est donnée par l'équation (40) :

$$(40) \quad f(P) = \frac{d_{cs}(P, S)}{R_{cs}(P, S)} - 1$$

Cette équation ressemble à la formule (39) de f_{cyl} donnée plus haut, excepté que les fonctions d_{cs} et R_{cs} doivent être définies.

Avec les symboles utilisés dans la Figure 65, la fonction $d_{cs}(P, S)$ renvoie la distance du point P à P_P ; le point P_P étant la projection de P sur le segment S (segment $[P_1P_2]$) suivant la direction perpendiculaire à la droite reliant P à la pointe P_c du cône. La zone grisée représente la partie « tronc de cône » du cône-sphère basé sur S . Si P_P est en dehors de cette zone (voir les exemples $P_{P'}$ et $P_{P''}$ dans la Figure 65), alors la distance de P à P_1 (respectivement P_2) est utilisée si P_P est plus proche de P_1 que de P_2 (respectivement plus proche de P_2 que de P_1).

La fonction $R_{cs}(P, S)$ renvoie le rayon du cône-sphère à la position correspondant à P . Ce rayon vaut R_1 (respectivement R_2) si P_P est en dehors de la zone grisée et plus proche de P_1 que de P_2 (respectivement plus proche de P_2 que de P_1). Si P (et donc P_P) est dans la zone grisée, alors le rayon du cône-sphère est la distance de P_P à l'intersection P_t de tangente au cône-sphère partant de P_c , avec sa droite perpendiculaire passant par P_P .

Définissons donc maintenant les équations (41) à (46) permettant de calculer tous les symboles de la Figure 65 entrant dans la formule de $f(P)$. Connaissant P_1 , P_2 , R_1 et R_2 qui sont les coordonnées des extrémités de chaque segment du squelette et les rayons des branches à ces extrémités, on peut commencer par trouver le point P_c grâce à :

$$(41) \quad P_c = P_1 + \frac{R_1}{R_1 - R_2} \overrightarrow{P_1P_2}$$

De plus, pour chaque point P , avec Δ étant la ligne de support du segment $[P_1, P_2]$ (c'est-à-dire l'axe du cône-sphère), nous pouvons calculer les angles et les distances Euclidiennes entre deux points (notées $dist(point1, point2)$) suivantes :

$$(42) \quad \alpha = \arcsin \frac{R_1}{\text{dist}(P_1, P_c)}$$

$$(43) \quad \beta = \arcsin \frac{\text{dist}(P, \Delta)}{\text{dist}(P, P_c)}$$

$$(44) \quad \text{dist}(P, P_p) = \text{dist}(P, P_c) \times \tan \beta$$

$$(45) \quad \text{dist}(P_p, P_c) = \sqrt{\text{dist}(P, P_c)^2 + \text{dist}(P, P_p)^2}$$

$$(46) \quad \text{dist}(P_p, P_t) = \text{dist}(P_p, P_c) \times \sin \alpha$$

Enfin, nous pouvons définir les deux fonctions d_{cs} et R_{cs} nécessaires à la définition de la fonction de potentiel f , par les équations (47) et (48) :

$$(47) \quad d_{cs}(P, S) = \begin{cases} \text{dist}(P, P_1) & \text{si } \overrightarrow{P_p P_1} \cdot \overrightarrow{P_p P_2} \geq 0 \quad \text{et} \quad \text{dist}(P_p, P_1) < \text{dist}(P_p, P_2) \\ \text{dist}(P, P_p) & \text{si } \overrightarrow{P_p P_1} \cdot \overrightarrow{P_p P_2} < 0 \\ \text{dist}(P, P_2) & \text{si } \overrightarrow{P_p P_1} \cdot \overrightarrow{P_p P_2} \geq 0 \quad \text{et} \quad \text{dist}(P_p, P_2) < \text{dist}(P_p, P_1) \end{cases}$$

$$(48) \quad R_{cs}(P, S) = \begin{cases} R_1 & \text{si } \overrightarrow{P_p P_1} \cdot \overrightarrow{P_p P_2} \geq 0 \quad \text{et} \quad \text{dist}(P_p, P_1) < \text{dist}(P_p, P_2) \\ \text{dist}(P_p, P_t) & \text{si } \overrightarrow{P_p P_1} \cdot \overrightarrow{P_p P_2} < 0 \\ R_2 & \text{si } \overrightarrow{P_p P_1} \cdot \overrightarrow{P_p P_2} \geq 0 \quad \text{et} \quad \text{dist}(P_p, P_2) < \text{dist}(P_p, P_1) \end{cases}$$

Il est possible d'obtenir des volumes approchant avec la définition de Tigges et Wyvill [121] ou avec une surface de convolution comme on peut le constater dans la thèse de Sherstyuk [110], mais seule la méthode présentée ici permet de représenter la forme exacte d'un cône-sphère implicite.

3.2.2 Adoucissement des jonctions par mélange de potentiels (« *blending* »)

Le choix des surfaces implicites pour représenter les segments de branches a été motivé par leur principal avantage qui est leur possibilité de mélange (« *blending* ») pour en adoucir les jonctions. Cependant, comme nous en avons discuté au paragraphe 3.1, les méthodes qui ont été proposées jusqu'alors pour adoucir la jonction entre des objets implicites à base de squelette sans générer de gonflement étaient prévues pour des surfaces

lisses et ne sont donc pas bien adaptées au déplacement de surface en vue de « sculpter l'écorce dans le bois ».

D'autre part certaines de ces méthodes ne parviennent pas complètement à supprimer le gonflement mais seulement à le réduire. C'est notamment le cas des méthodes étudiées par Bloomenthal [15] [16]. En revanche, les surfaces de convolution [14] [111] fonctionnent très bien pour joindre deux segments mais, au-delà, le gonflement réapparaît. Nous avons donc cherché une méthode qui offre la possibilité de :

- mélanger plusieurs (au moins deux) objets implicites cylindriques basés chacun autour d'un segment ;
- modifier la fonction de potentiel pour déplacer la surface fidèlement à partir d'une carte de hauteurs.

Le mélange sans gonflement de plus de deux objets implicites de diamètres proches est difficile. Afin de simplifier le problème, nous avons choisi de mélanger les segments de branches deux par deux et de prendre l'union de ces mélanges comme résultat. Dans le cas général d'un embranchement simple, impliquant trois segments (un parent et deux enfants), le résultat est identique à ce que devrait être le mélange des trois segments car Holton [55] a fait remarquer que les branches d'un tel embranchement sont coplanaires. En revanche, dans le cas moins courant d'un embranchement à plus de deux branches filles, un léger creux apparaîtra entre les branches filles mais il n'est pas gênant visuellement, surtout avec les textures d'écorce en relief que nous appliquons (cf. paragraphe 3.3). La Figure 66 illustre ces deux cas.

Toutefois, le fait de mélanger les segments de branches deux à deux impose que leur longueur soit au moins deux fois supérieure au rayon du champ de potentiels, comme le montre la Figure 67, et même encore plus dans le cas d'angles aigus entre deux segments. Il faut donc définir une méthode de mélange qui puisse se contenter d'un champ de potentiel réduit tout en répondant aux critères cités plus haut. Les surfaces de convolution offrent, certes, un mélange de surfaces sans gonflement pour joindre deux segments de branche, mais le mélange est trop fort et nécessiterait des segments de branche très longs pour éviter que le mélange à une extrémité d'un cône-sphère n'interfère avec le mélange à l'autre extrémité. Or, ceci peut être contraignant pour modéliser précisément des branches.

D'autre part, les surfaces de convolution ne représentent pas exactement le cône-sphère mais un volume approchant, mais ce n'est pas vraiment gênant car ce qui compte n'est pas de respecter exactement la forme des primitives mais qu'elles permettent au final d'obtenir un rendu de branches jointes de façon douce.

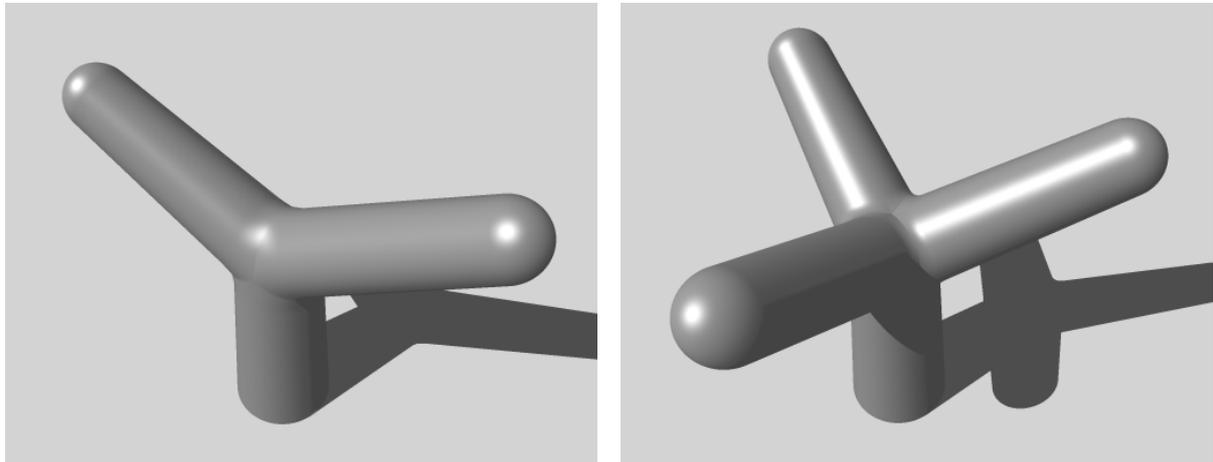


Figure 66 : Embranchements de surfaces implicites mélangées. A gauche : embranchement simple. A droite : embranchement à plus de deux branches filles.

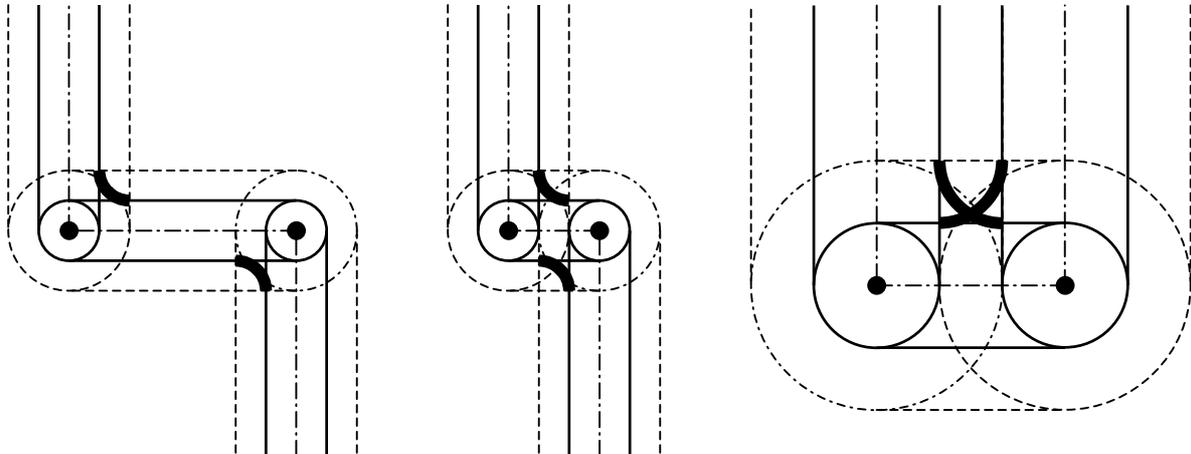


Figure 67 : Contrainte de longueur minimale des segments par rapport au rayon du champ de potentiels. A gauche la longueur des segments est très largement suffisante. Au centre : la longueur devient critique mais les zones de mélange n'étant pas du même côté, il n'y a pas de problème. A droite : il y a un problème d'intersection des zones de mélange quand elle est située du même côté d'un segment trop court.

L'équation (40) donne le potentiel d'un point P par rapport à un cône-sphère. Le mélange de deux cônes-sphères se fait en mélangeant les potentiels de P par rapport à chacun d'eux, ce qui donne le potentiel de P par rapport à l'ensemble de ces deux cônes-sphères. Dans un premier temps, si l'on ne se soucie pas du gonflement, Bloomenthal [15] a proposé une formule simple, en se basant sur les travaux de Rockwood [105]. Soit Pot_1 et Pot_2 le potentiel de P respectivement par rapport aux segments de branche basés autour des segments de squelette S_1 et S_2 , l'équation (49) définit le mélange de ces potentiels par :

$$(49) \quad B(Pot_1, Pot_2) = 1 - (1 - h(Pot_1))(1 - h(Pot_2))$$

La fonction h est une fonction de mélange, de forme sigmoïde décroissante, qui rend les potentiels non linéaires afin de les mélanger doucement. En effet des potentiels linéaires génèreraient un mélange à l'intérieur du coude comme celui de gauche de la Figure 68.

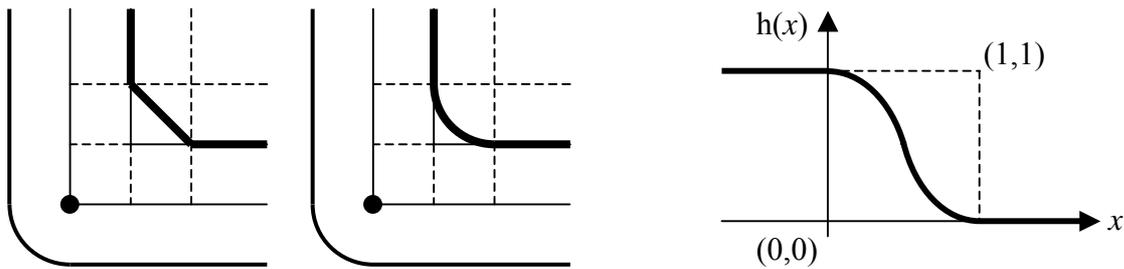


Figure 68 : Importance de la non-linéarité des potentiels. A gauche : les potentiels sont linéaires. Au centre : ils sont transformés par la fonction h (à droite).

Wyvill et al. ont défini cette fonction h par l'équation (50) :

$$(50) \quad h(x) = \begin{cases} 1 & \text{si } x \leq 0 \\ \frac{9 - 4x^6 + 17x^4 - 22x^2}{9} & \text{si } 0 < x < 1 \\ 0 & \text{si } x \geq 1 \end{cases}$$

Nous avons constaté qu'elle est égale à l'équation (51) qui peut être plus rapide à calculer :

$$(51) \quad h(x) = \begin{cases} 1 & \text{si } x \leq 0 \\ 0,5 + \frac{\cos(x\pi)}{2} & \text{si } 0 < x < 1 \\ 0 & \text{si } x \geq 1 \end{cases}$$

Il est à noter que le mélange $h(x)$ est le plus équilibré lorsque x vaut 0,5. Or, x est le potentiel à mélanger. Nous redéfinissons donc l'isosurface de façon à ce qu'elle soit définie pour $f(P) = 0,5$. Il faut donc modifier artificiellement les rayons R_1 et R_2 en les remplaçant respectivement par $R_1' = R_1/1,5$ et $R_2' = R_2/1,5$ dans l'équation (48).

L'équation (49) effectue le mélange brut des potentiels, ce qui génère un gonflement comme celui que l'on peut observer dans l'image de gauche de la Figure 63. Si l'on considère le problème en deux dimensions dans le plan $P_{S_1S_2}$ formé par S_1 et S_2 , il faut que le volume soit l'union des cônes-sphères à l'extérieur des coudes et le mélange à l'intérieur. En trois dimensions, le volume est toujours formé par l'union U des deux cônes-sphères, donnée par $U = \max(h(Pot_1), h(Pot_2))$, à l'extérieur des coudes et il équivaut bien au mélange à l'intérieur du coude sur le plan $P_{S_1S_2}$ mais, entre ce plan et sa normale, il faut interpoler, en fonction de P , le volume ajouté pour adoucir la jonction sans gonflement indésirable.

Bloomenthal [15] et Rockwood [105] ont utilisé une mesure de convexité de la surface qui n'est malheureusement pas compatible avec les cônes-sphères qui introduisent plusieurs cas particulier et qui, selon la définition de Bloomenthal, génère tout de même un léger gonflement à l'extérieur des coudes dans certains cas. Nous avons donc dû redéfinir cette mesure notée *Convex*.

Soit \vec{v}_p le vecteur normé donnant la direction de la projection de P sur $P_{S_1S_2}$ à partir du point de jonction P_j de S_1 et S_2 et soient \vec{v}_1 et \vec{v}_2 les vecteurs normés donnant respectivement la direction des segments S_1 et S_2 à partir de P_j , on calcule la proportion de l'angle (\vec{v}_1, \vec{v}_p) par rapport à l'angle (\vec{v}_1, \vec{v}_2) avec la formule (52) :

$$(52) \quad prop_{angle} = 1 - \frac{angle(\vec{v}_1, \vec{v}_p)}{angle(\vec{v}_1, \vec{v}_2)}$$

On calcule ensuite les projections P_{S_1} et P_{S_2} de P , respectivement sur les segments (pas les droites) S_1 et S_2 et on calcule le vecteur \vec{v}_b qui interpole les vecteurs $\overrightarrow{P_{S_1}P}$ et $\overrightarrow{P_{S_2}P}$ pondérés avec $prop_{angle}$ au moyen de la formule (53) :

$$(53) \quad \vec{v}_b = \overrightarrow{P_{S_1}P} \times prop_{angle} + \overrightarrow{P_{S_2}P} \times (1 - prop_{angle})$$

Il ne reste plus qu'à normaliser ce vecteur et le projeter sur le plan $P_{S_1S_2}$. La norme au carré, notée C du vecteur résultant permet de calculer la mesure de convexité de la surface par rapport à P avec la formule (54) :

$$(54) \quad Convex = 1 - C \left(1 - \frac{angle(\vec{v}_1, \vec{v}_2) - \alpha_c}{\pi - \alpha_c} \right)$$

Nous avons également introduit $\alpha_c = 3\pi/4$ qui est l'angle critique auquel il faut rapporter l'angle entre les segments S_1 et S_2 pour éviter un gonflement lorsque cet angle est supérieur à $3\pi/4$.

Enfin, tandis que le potentiel de P à l'extérieur d'un coude vaut U , on peut calculer le potentiel mélangé de P à l'intérieur d'un coude, sans gonflement, grâce à l'équation (55), avec $b = B(Pot_1, Pot_2)$:

$$(55) \quad Potentiel(P) = b + Convex(U - b)$$

Comme on peut le remarquer sur la Figure 66, notre mesure de convexité n'est pas parfaite et génère de légers défauts de surface. Les résultats sont néanmoins acceptables sachant que la surface des branches sera recouverte d'une texture d'écorce en relief qui fera disparaître ces problèmes. D'autant plus qu'une branche n'est pas censée être parfaitement cylindrique et que nous rajoutons même volontairement du bruit à basse fréquence au paragraphe 3.3.2.

Notons qu'il serait possible d'utiliser les méthodes de mélange prévues pour des surfaces lisses en polygonisant ces surfaces puis en déplaçant la surface des triangles générés suivant une carte de déplacement par rapport au squelette pour ajouter a posteriori le relief de l'écorce. Mais cette méthode obligerait, soit à mémoriser toutes les branches

polygonisées avant le rendu, ce qui risquerait de consommer énormément de mémoire, soit à repolygoniser chaque segment de branche (et ses voisins) à chaque test d'intersection avec un rayon, ce qui prendrait sans doute plus de temps que le « *ray marching* », sauf éventuellement en profitant d'un cache [93] pour stocker temporairement quelques-uns des derniers segments polygonisés. Avec un certain nombre d'optimisations, cette méthode peut donc être une alternative intéressante à explorer.

3.3 Application de la texture d'écorce

Nous avons présenté au Chapitre 2 une méthode pour synthétiser une texture infinie d'écorce donnant une couleur et une élévation pour n'importe quelles coordonnées cylindriques (u,v) . Or, la surface d'un arbre est plus complexe qu'un simple cylindre. Nous présentons donc, au paragraphe 3.3.1, une méthode simple pour calculer les coordonnées (u,v) correspondant à n'importe quel point de la surface de l'arbre.

Comme nous représentons les branches par surfaces implicites qui doivent être rendues par « *ray marching* », en calculant le potentiel de points de l'espace sur le trajet des rayons pour déterminer s'ils sont sur la surface et, comme cette surface est justement dépendante du relief de la texture, cette méthode doit également nous permettre de déterminer les coordonnées de texture correspondant à n'importe quel point de l'espace, qu'il soit à la surface des branches ou non, de manière à ce que ces coordonnées ne dépendent pas de la surface mais que ce soit l'inverse. Ceci nous permet alors également, au paragraphe 3.3.2, de déplacer la surface pour représenter la texture d'écorce en relief grâce à l'élévation renvoyée par $f_{tex}(u,v)$.

3.3.1 Paramétrisation des coordonnées de texture

Nous partons de l'hypothèse que les branches sont modélisées par un squelette formé de segments définissant implicitement un volume à base de cône-sphères (un par segment du squelette). Pris indépendamment, les segments de branches ont donc une forme plutôt cylindrique ; mais, pour représenter l'arbre, ils sont disposés bout à bout pour former des coudes (deux segments joints) et des embranchements (trois segments ou plus). D'autre part, en raison de certaines contraintes dues à notre méthode de mélange de volumes

implicites, nous ne considérons que deux types de diamètres de segments de branches : les branches qui conservent leur diamètre d'un segment à l'autre et les petites branches dont le diamètre est la moitié du segment de branche parent.

Tandis que le problème d'application de texture sur les coudes est assez facile à gérer en comprimant la texture à l'intérieur des coudes et en l'étirant à l'extérieur (voir Figure 69) pour qu'elle suive le long des branches de manière « fluide », le problème des embranchements est plus délicat. Il semble en effet plus difficile de représenter la séparation de la texture à un embranchement. Cependant, en observant les arbres dans la nature, nous avons constaté que la nature elle-même n'a pas résolu ce problème sans marquer une discontinuité au point de séparation des branches, comme le montre distinctement la Figure 25, même si la discontinuité n'est pas toujours aussi visible avec toutes les écorces.

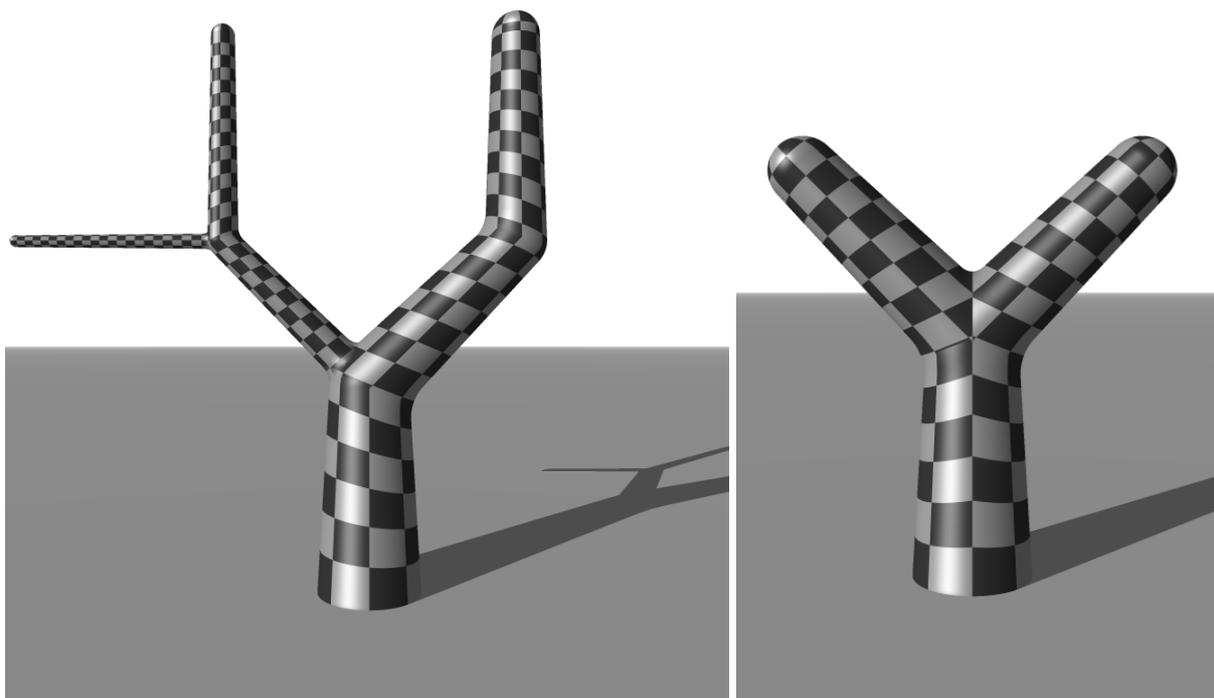


Figure 69 : Paramétrisation de la texture sur des branches. La texture suit les branches de même diamètre mais elle est mélangée à la jonction de branches de tailles différentes. Sur l'image de droite, on remarque la discontinuité de texture à l'intérieur de l'embranchement. Elle devient toutefois beaucoup moins visible avec une texture d'écorce rugueuse.

La méthode que nous employons préserve la continuité le long des branches mais elle introduit une discontinuité à l'intérieur des embranchements (voir Figure 69), du moins dans le cas où les branches auraient le même diamètre ; car, dans le cas contraire, nous interpolons la texture entre la grosse branche et la petite. Cette interpolation permet d'ailleurs, si on le souhaite, d'assigner une texture différente aux branches qui partent d'un tel embranchement puisqu'on ne risquera pas de voir de discontinuité brusque de la texture. Pour faire partir de petites branches à partir de plus grosses sans interpoler, il faudrait utiliser un cône-sphère dont le grand rayon serait celui de la grosse branche dont il est issu et le petit rayon serait égal au rayon que doit avoir la petite branche.

Notre méthode de paramétrisation de la texture étant basée sur le squelette, il nous faut savoir exactement quelle branche a été intersectée par le rayon pour déterminer les coordonnées (u,v) en fonction de la position du point d'intersection P par rapport au segment. Ce point pouvant appartenir à plusieurs cône-sphères, puisque ceux-ci sont joints, nous isolons chaque segment de branche par les plans médians formés par les angles entre deux segments de branches jointes, comme l'illustre la Figure 70 ; mais seulement pour les branches de même diamètre car les petites branches doivent être mélangées. Le plan de base des petites branches issues de plus grosses branches sera donc perpendiculaire à l'axe du segment afin de minimiser les distorsions puisqu'elles sont inutiles dans ce cas.

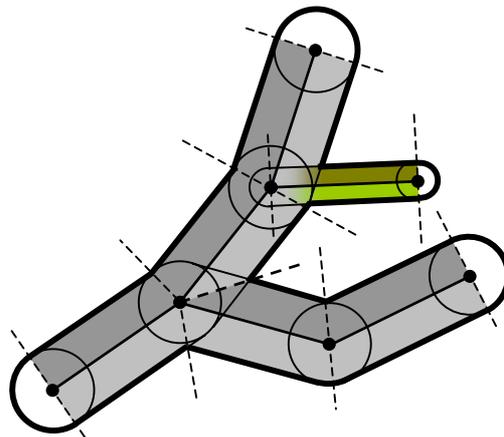


Figure 70 : Isolement de chaque segment de branche par des plans médians (en pointillés). Les petites branches (ici en vert) sont simplement mélangées. Les zones en gris clair et gris foncé montrent la continuité de la texture le long des branches à l'exception de l'intérieur d'un embranchement de branches de même diamètre.

De plus, comme le diamètre des branches rétrécit (car elles sont de forme conique) il faut adapter la texture pour qu'elle garde le même rapport largeur/hauteur. En d'autres termes, il faut la « tasser » au rythme où les branches rétrécissent. Le calcul des coordonnées u et v étant indépendant, nous le décomposons dans les deux paragraphes suivants.

3.3.1.1 Calcul de la coordonnée u

Pour calculer la coordonnée u d'un point P dans le repère du cône-sphère, il faut tout d'abord connaître le nombre d'échantillons nécessaires N_{ETB} pour faire le tour d'une branche. Cette valeur est égale à la largeur L_{table} de la table que nous décrivons au paragraphe 2.3.2. Soit ω l'angle égal à $2\pi/N_{ETB}$ et soit α défini l'équation (56) comme étant l'angle correspondant à $P(x_P, y_P, z_P)$ autour de l'axe z dans le repère du cône-sphère, nous obtenons la coordonnée u par l'équation (57).

$$(56) \quad \alpha = \frac{y_P}{|y_P|} \times \arccos\left(\frac{x_P}{\sqrt{x_P^2 + y_P^2}}\right)$$

$$(57) \quad u = (\alpha \bmod \omega) \times \frac{L_E}{\omega}$$

Cependant, pour que le calcul de u soit correct, il faut qu'aucune rotation d'un angle différent de ω (dépendant de N_{ETB}) n'ait été effectuée autour de l'axe z sur un couple de cône-sphères consécutifs et de même diamètre à leur jointure. La Figure 71 illustre ce problème. Néanmoins, cette contrainte ne se pose pas pour des cône-sphères successifs de rayons différents car, dans ce cas, la texture est mélangée et non continue à leur jointure. Il est toutefois possible de contourner cette contrainte de deux façons. La première permet d'obtenir sensiblement le même résultat qu'en utilisant une rotation autour de l'axe z car elle est, en pratique, le plus souvent suivie ou précédée d'une rotation autour de l'axe x ou y . Il est donc possible de combiner ces deux rotations pour orienter l'axe z (qui est celui qui importe le plus puisqu'il donne la direction des segments de branche), même si les axes x et y résultants sont différents. L'autre façon, plus simple et qui permet la rotation autour de l'axe z , est d'ajouter ou soustraire l'angle de rotation (répercuté le long des branches) autour de l'axe z à l'angle α dans l'équation (56).

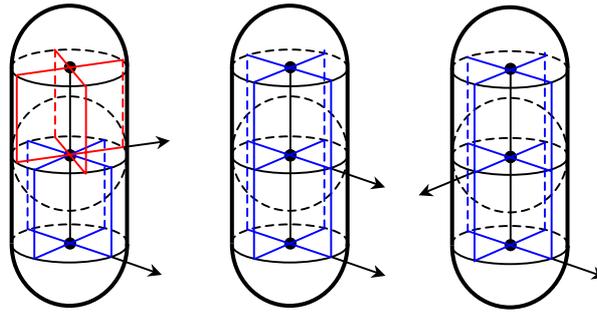


Figure 71 : Lors de la modélisation du squelette de l'arbre, la rotation autour de l'axe z de deux segments (cône-sphères) consécutifs n'est pas autorisée si elle est différente de $2\pi/N_{ETB}$ (dans cet exemple, $N_{ETB} = 4$). A gauche : cas interdit. Au centre et à droite : cas autorisés. Les flèches montrent l'axe x utilisé comme base pour l'angle α donnant la position de P autour de l'axe z .

3.3.1.2 Calcul de la coordonnée v

Le calcul de la coordonnée v est plus compliqué car il gère deux problèmes. Ce sont, d'une part, la contraction et l'étirement de la texture aux jonctions de segments de branches non colinéaires et, d'autre part, le tassement de la texture au fur et à mesure que le diamètre des branches rétrécit pour respecter son rapport largeur/hauteur.

Contraction et étirement de la texture aux jonctions

Supposons dans un premier temps que nous n'ayons pas à gérer le rapport largeur/hauteur de la texture. D'après la Figure 70, tous les points situés sur un même plan médian, entre deux segments de branche, sont censés avoir la même coordonnée v . Nous considérons, comme base d'une branche, un nœud du squelette de l'arbre où une petite branche est issue d'une plus grosse (cf. Figure 72). Une branche est donc une succession de segments de branche dont les diamètres concordent à leurs jonctions. Le diamètre à la base des branches est noté $Diamètre_{base}$. La distance (notée $Distance_{base}$) de la base d'une branche au point P_1 (première extrémité) d'un segment de cette branche – ainsi qu'à tous les points sur un plan médian passant par P_1 – est donc la somme des longueurs des segments entre la base de la branche et P_1 . La base du tronc (racine) est aussi considérée comme une base de branche.

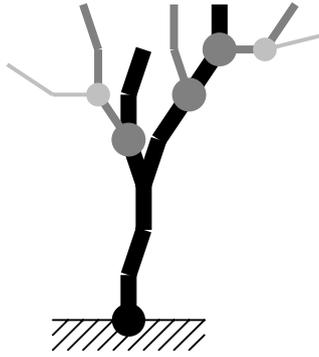


Figure 72 : Les sommets représentant des bases de branches (à la naissance des branches de diamètre inférieur à la branche parente) sont représentés ici par des points circulaires. Les branches de niveau 1 sont en noir et leur base est le point noir (unique) à la base du tronc. Les branches de niveau 2 sont en gris foncé et celles de niveau 3 en gris clair.

Pour chaque point P entre deux plans consécutifs, situé à une distance d_1 (respectivement d_2) du plan passant par le sommet P_1 (respectivement P_2), v sera calculée à partir de la distance d_1 proportionnellement à la longueur L du segment par rapport à la distance $l = d_1 + d_2$ des projections P_{P_1} et P_{P_2} de ce point P sur les deux plans, comme le montre la Figure 73.

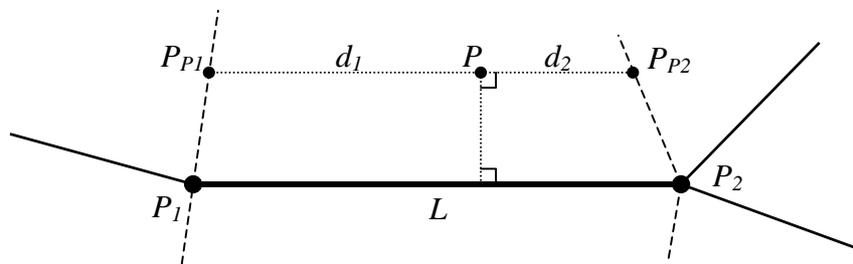


Figure 73 : Déformation de la texture grâce à des plans médians séparant le segment en trait gras de ses voisins.

Il est important de noter que les distances d_1 et d_2 sont les distances minimales de celles trouvées avec tous les plans médians. Plus précisément, pour trouver d_1 , on détermine le plan médian entre le segment S étudié et tous les segments ayant une extrémité commune avec le point P_1 (c.-à-d. segment parent et segments frères) et un rayon identique en ce point ; puis, on projette P dans la direction de l'axe sur chacun de ces plans

pour obtenir un ensemble de points dont P_{P1} est un exemple sur la Figure 73. Enfin d_1 est la distance minimale entre P et chacun de ces points. Puis on fait de même pour déterminer d_2 avec les segments ayant une extrémité commune à P_2 et un rayon identique en ce point. Dans le calcul de d_1 , les branches avec un rayon inférieur à leur parent sont traitées plus simplement en prenant toujours le plan perpendiculaire à leur axe. Il est à noter que la projection suivant la direction de l'axe peut également se faire suivant la direction de la droite reliant P à P_c (voir Figure 65) puis nous traitons de cône-sphères et non de cylindres mais la différence est négligeable.

La distance « corrigée » Z serait alors la distance de P_1 à la base de la branche plus la distance d_1 comprimée ou étirée par le ratio L/l , d'où la formule (58) :

$$(58) \quad Z = Distance_{base} + \frac{d_1 \cdot L}{d_1 + d_2}$$

L'équation (59) donne la taille des pixels, à la base de la branche, en unités de distance dans l'espace où est défini l'arbre :

$$(59) \quad taille_{pixel} = \frac{\pi \times Diamètre_{base}}{N_{ETB} \times L_E}$$

Soit v' la coordonnée verticale de texture ne prenant pas en compte le rapport largeur/hauteur de la texture, cette coordonnée est égale à la distance Z transformée en pixels en se basant sur la taille des pixels à la base de la branche, d'où la formule :

$$(60) \quad v' = \frac{Z}{taille_{pixel}} = Z \times \frac{N_{ETB} \times L_E}{\pi \times Diamètre_{base}}$$

Maintient du rapport largeur/hauteur de la texture

Il faut maintenant tasser la texture en augmentant la coordonnée Z définie ci-dessus pour que le rapport entre la largeur et la hauteur de la texture reste constant le long des branches afin que la texture ne paraisse pas étirée sur les branches les plus fines. Il est cependant nécessaire de connaître la façon dont évolue le diamètre des branches. On peut différencier deux formes principales de variation du diamètre des branches pouvant être choisies lors de la modélisation des branches :

- soit ce diamètre rétrécit d'un certain facteur $f_{rétréc}$ à chaque fois que l'on avance d'une unité de longueur le long des branches (i.e. $D_{i+1} = D_i/f_{rétréc}$) ;
- soit il rétrécit de façon linéaire en diminuant à chaque unité de $Diamètre_{base}/L_{branche}$ (rapport entre le diamètre à la base de la branche et la longueur de celle-ci) jusqu'à finir en pointe (i.e. $D_{i+1} = D_i - Diamètre_{base}/L_{branche}$).

Dans le premier cas – qui est celui que nous avons choisi – la coordonnée v est calculée par l'équation (61) dont la partie la plus à droite permet de tasser la texture :

$$(61) \quad v = Z \times \frac{N_{ETB} \times L_E}{\pi \times Diamètre_{base}} \times \left(\frac{1 + f_{rétréc}}{2} \right)^Z$$

Dans le second cas, qui est néanmoins tout à fait utilisable, la coordonnée verticale selon une diminution linéaire du diamètre des branches est calculée par l'équation (62) :

$$(62) \quad v_{linéaire} = Z \times \frac{N_{ETB} \times L_E}{\pi \times Diamètre_{base}} \times \left(1 - \frac{x}{L_{branche}} \right)$$

3.3.1.3 Remarques

En ce qui concerne les petites branches issues d'une plus grosse branche, il est possible de leur appliquer une autre texture que celle de la branche parente puisque le problème de discontinuité ne se pose pas, étant donné que la texture de la petite branche est mélangée avec celles de sa branche parente et de ses sœurs. Ceci permet par exemple d'utiliser une texture très rugueuse pour le tronc et les branches principales, une texture un peu plus lisse pour les branches secondaires, et une autre texture pour les rameaux.

Enfin, la proportion $mélange_{texture}$ de texture (variant de 1 à 0) de la petite branche par rapport à la grosse en P vaut $Pot_1/(Pot_1+Pot_2)$. De cette façon, la texture de la petite branche s'estompe au fur et à mesure que P s'approche du gros segment de branche et s'éloigne du petit. Soient $c_{PetiteBranche}$ et $c_{GrosseBranche}$ les couleurs correspondant à P respectivement sur la petite et sur la grosse branche, on calcule donc la couleur c à afficher grâce à la formule (63) :

$$(63) \quad c = mélange_{texture} \times c_{PetiteBranche} + (1 - mélange_{texture}) \times c_{GrosseBranche}$$

3.3.2 Déplacement de surface (« *displacement mapping* »)

Nous disposons d'une fonction qui nous donne le potentiel de n'importe quel point P de l'espace par rapport à la surface d'un segment de branche mélangé avec ses voisins. En modifiant ce potentiel, il est donc possible de déplacer la surface par rapport au point. Nous avons également défini les fonctions pour calculer les coordonnées de texture correspondant à n'importe quel point de l'espace. A partir de celles-ci nous pouvons obtenir une couleur et surtout une élévation.

En soustrayant cette élévation au potentiel de P , nous le faisons diminuer. Or, nous avons ce potentiel de façon à ce qu'il soit décroisse en se rapprochant du squelette des branches. Donc, en faisant diminuer P , on simule son rapprochement à la surface.

Cependant, pour pouvoir mélanger les potentiels harmonieusement, nous avons dû les rendre non-linéaires. Il faut donc transformer la profondeur à creuser pour qu'elle corresponde avec le potentiel. La formule de la valeur à soustraire au potentiel est donnée par l'équation (64) :

$$(64) \quad \text{déplacement}_{\text{texture}} = 0,5 \times \sin\left(\text{Prop}_{\text{ProfondeurEcorce}} \times 3\pi / 2\right) \times (1 - \text{Hauteur}(u, v))$$

De la même manière, il est possible de soustraire une valeur aléatoire pour réduire l'aspect rectiligne des branches. Mais il faut faire attention à ne jamais retirer plus d'un tiers du rayon de la branche car le champ de potentiel qui fait 1,5 fois le rayon de la branche ne le permet pas. Mais ce n'est pas gênant puisque nous n'avons rencontré aucune écorce qui ait une profondeur supérieure à un dixième du rayon des branches.

Avec quelques modifications, le rendu de notre représentation pourrait être accéléré par la méthode de Smits [114] qui permet de faire un rendu par lancer de rayons d'un objet modélisé sous forme de triangles auxquels on ajoute une texture par déplacement de surface. En effet les conditions requises par cette méthode sont remplies par notre représentation de l'écorce sur les branches :

- il n'y a pas d'auto-intersections de la surface ;
- l'intervalle de déplacement est borné et assez limité (les craquelures de l'écorce ne sont pas très profondes, même pour les écorces les plus rugueuses) ;

- chaque point correspond à une seule position et valeur de déplacement.

La seule différence – de taille, cependant – entre cette méthode et la nôtre est que nous n’employons pas de triangles pour représenter nos segments de branches mais il serait tout à fait possible de trianguler les segments de branches lisses avec une méthode dite de « *marching cubes* », ce qui formerait un nombre limité de triangles, surtout si l’on utilise une méthode adaptative suivant la taille et la distance par rapport à la caméra de chaque segment de branche.

3.4 Application par distribution d’éléments géométriques

Les éléments de texture de forme particulière, tels que les épines qui dépassent de certaines écorces, ne peuvent pas être traités comme le reste de l’écorce pour plusieurs raisons :

- Notre méthode d’extraction de relief à partir d’une photographie donne des résultats visuellement cohérents mais approximatifs et des motifs de formes aussi reconnaissables que des épines risqueraient d’être trop déformés ;
- Les épines sont assez hautes et risquent donc de générer des ombres trop importantes ;
- Si l’on devait « creuser » le relief de l’écorce en incluant les épines, leur hauteur nécessiterait d’augmenter le volume englobant pour rajouter principalement du vide et ralentirait considérablement la méthode d’approche de la surface par « *ray marching* » (cf. paragraphes 1.3.3.5 et 3.2) qui est déjà assez lente.

Il est donc préférable d’appliquer uniquement la texture sans les épines et d’ajouter ces dernières comme des éléments géométriques à part. La Figure 74 illustre un des résultats de la méthode généraliste que nous avons proposée [33] ; mais il existe beaucoup d’autres méthodes ayant le même but. En fait, la plupart des méthodes permettent souvent d’appliquer, soit des éléments géométriques [39], soit des morceaux de texture 2D [96], soit les deux, comme c’est le cas pour notre méthode [33], même si nous n’utilisons, pour l’écorce, que le cas des éléments géométriques de cette méthode puisque le reste de la texture est traité autrement au paragraphe 3.3 afin d’être particulièrement adapté à la morphologie des branches.

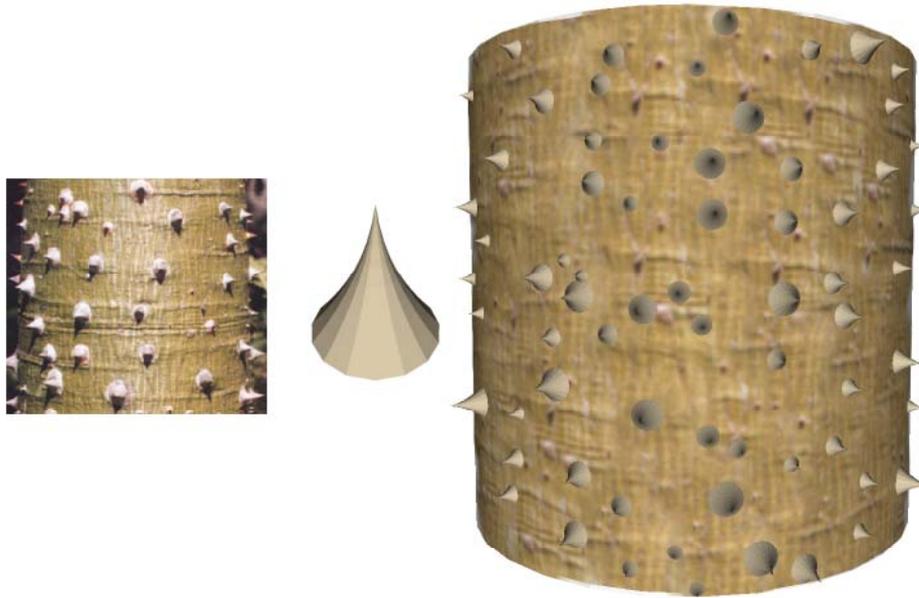


Figure 74 : Répartition d'épines sur un tronc.

La méthode [33] a été conçue pour appliquer des textures sur des surfaces définies par un maillage de facettes. Avant le rendu, on discrétisera donc les plus grosses branches, sans texture (donc lisses), qui sont celles sur lesquelles on appliquera les épines, ce qui permet d'économiser un peu de mémoire en ne prenant pas en compte les plus petites branches.

A la base, cette méthode analyse le modèle en le segmentant pour isoler les particules (éléments de texture) à dupliquer et détermine la distance qui sépare chaque particule de ses voisins en cherchant les voisins les plus proches jusqu'à en trouver au moins un dans chacune des quatre directions diagonales. Toutefois, dans le cas des épines, elles sont disposées aléatoirement sur la surface et il suffit d'indiquer une distance minimale $DistMin_{particules}$ et une distance maximale $DistMax_{particules}$ entre les épines.

Le processus simplifié d'arrangement de ces particules de texture est donc le suivant. On commence par placer une première particule (épine) sur la surface. Pour cela, on choisit aléatoirement un sommet du maillage et on y place une épine en l'orientant de façon à ce qu'elle soit alignée verticalement avec la normale à la surface en ce point (moyenne des normales des facettes dont ce point est un sommet, pondérées par l'aire de chaque facette). On dispose ensuite, aléatoirement, une épine autour de cette « graine » dans une direction

aléatoire à une distance comprise dans l'intervalle de distance défini au début, tout en vérifiant que la nouvelle particule n'est pas trop proche d'une autre particule précédemment ajoutée. On se servira de la topologie du maillage pour calculer cette distance de facette en facette. De la même manière que la distance séparant les épines est aléatoire (dans un certain intervalle), on peut faire varier la taille de chaque épine, y compris par rapport au diamètre de la branche à laquelle est rattachée la facette supportant l'épine.

Le processus d'insertion de nouvelles particules se poursuit jusqu'à ce qu'on ne puisse plus insérer de nouvelles particules. Cette notion n'est cependant pas facile à déterminer car une tentative d'insertion de particule infructueuse ne signifie pas qu'il n'est plus possible d'insérer de nouvelles particules. C'est pourquoi nous proposons d'assigner, à chaque facette (triangulaire) du maillage, un nombre maximal de tentatives infructueuses d'insertion de nouvelles particules dépendant de la distance moyenne des particules et de l'aire de chaque facette f_i grâce à la formule (65) :

$$(65) \quad NbMaxTentatives(f_i) = \min\left(1, \frac{aire(f_i)}{(DistMin_{particules} + DistMax_{particules})/2}\right)$$

A chaque tentative sans succès d'insertion d'une particule sur une facette, on décrémentera ce compteur et, s'il devient nul, on supprimera la facette de la liste des facettes sur lesquelles on peut rajouter une particule. Le traitement s'arrêtera donc lorsque cette liste deviendra vide.

3.5 Conclusion

Nous avons présenté des méthodes permettant de représenter des branches basées sur un squelette d'arbre sous forme de segments. Ces branches sont constituées de cônes-sphères que nous définissons de façon implicite afin de pouvoir les mélanger et leur appliquer une texture en relief.

Nous avons proposé un moyen, qui est à notre connaissance le seul qui ne soit pas destiné qu'à des surfaces lisses, pour résoudre le problème de gonflement que l'on rencontre souvent en mélangeant des objets implicites. Cette méthode présente toutefois un

très léger défaut de surface à l'intérieur des coudes formés par deux cône-sphères successifs formant un angle aigu mais ce défaut ne se remarque pas une fois que l'on applique une texture d'écorce rugueuse. De plus, il apparaît à un endroit où la nature elle-même a du mal à générer une surface lisse. Cette méthode ne vient donc pas en remplacement de celles qui ont été développées pour les surfaces lisses mais en complément pour gérer les surfaces rugueuses sur des structures arborescentes à base de cône-sphères (ou d'autres formes implicites cylindriques).

Nous avons également conçu notre méthode de mélange de surfaces implicites de façon à pouvoir contrôler assez fidèlement la profondeur du relief issu des textures infinies que nous synthétisons au Chapitre 2. Nous avons donc aussi proposé une façon simple, rapide, efficace et proche de ce que nous avons pu observer dans la nature pour appliquer ces textures le long des branches sans effet périodique de répétition ni discontinuités visibles. Le seul endroit où de légères discontinuités apparaissent se situe encore une fois entre les branches « filles » d'un embranchement ; mais, là aussi, on observe une discontinuité au même endroit dans la nature.

Enfin, nous avons proposé une méthode pour répartir, sur la surface du tronc et des branches principales, des éléments de texture tels que des épines, qui ne pourraient pas être représentés par nos textures infinies en $2D\frac{1}{2}$.

Chapitre 4

Les feuilles

4.1 Introduction

Jusqu'ici nous n'avons traité que des branches et de l'écorce qui les recouvre. Il reste encore cependant un élément essentiel de l'arbre, qui apporte un poids substantiel au réalisme visuel : les feuilles.

Nous verrons dans ce chapitre quelle peut être la meilleure façon de modéliser des feuilles en vue d'un rendu réaliste ; comment disposer les feuilles sur l'arbre ; comment les afficher ; et enfin, par quels moyens l'on peut en améliorer encore le rendu grâce à la représentation des nervures pour leur donner du relief et influencer leur translucidité.

4.2 Représentation géométrique des feuilles

Comme nous l'avons vu en section 1.5, de nombreux travaux ont déjà été effectués pour synthétiser des feuilles. Cependant, bien que certains d'entre eux permettent d'obtenir de très bons résultats quant à la forme des feuilles [60] [71] [72] [98], il se pose alors le problème de leur texture qui, lorsqu'on observe les feuilles de près, ne peut pas être une simple couleur ou dégradé de couleurs.

Le meilleur moyen d'obtenir un résultat photo-réaliste est alors, logiquement, d'utiliser une photographie numérisée de feuille ; ce qui est, de plus, très facile à faire à l'aide d'un scanner à plat classique. Le fond de l'image de feuille obtenue est alors remplacé par une zone de transparence – soit au moyen d'une couleur spéciale, soit grâce à une carte de transparence (canal « alpha ») binaire ou plus précise – ce qui peut également être fait très facilement à l'aide d'un logiciel de traitement d'image en sélectionnant le fond (a priori blanc pour la plupart des scanners) avec une légère tolérance pour englober les

quelques variations du fond et surtout les contours de la feuille, puis en effaçant ce qui a été sélectionné pour le rendre transparent.

Enfin, on plaquera cette texture aux contours transparents sur une feuille modélisée plus ou moins grossièrement à l'aide de quelques polygones. Notons ici qu'il est tout de même possible d'utiliser une forme de feuille qui aura été modélisée à partir des méthodes de synthèse mentionnées précédemment, quitte à adapter manuellement les principaux sommets du maillage à la feuille numérisée. Le principal est que les polygones englobent totalement les parties opaques de la texture afin que les contours de ce qu'on verra de la feuille soient définis par la texture elle-même et non par les polygones comme le montre la Figure 33 du paragraphe 1.5.2. Cette méthode, utilisée entre autres par Bloomenthal [12], permet d'obtenir un contour parfait de la feuille à condition que la numérisation ait été effectuée avec une assez grande résolution.

En plus de l'accroissement du réalisme visuel apporté par la photographie, cette méthode permet de limiter considérablement le nombre de polygones nécessaires à la modélisation des feuilles, surtout sur les feuilles au contour complexe (dentelé, lobé...). Et quand on sait le nombre important de feuilles dans un arbre, on comprend aisément que le gain en temps d'affichage sera appréciable.

Dans l'absolu, un ou deux triangles pourraient suffire mais il convient d'affiner un peu plus le découpage si l'on veut pouvoir plier la feuille pour qu'elle ne soit pas complètement plate. Outre la courbure naturelle de la feuille tout entière, on peut noter que les principaux axes de pliure de la feuille sont situés sur les nervures primaires. Il sera donc judicieux d'aligner au mieux les nervures primaires avec des arêtes des polygones.

Cependant, on constate en observant une feuille que ses deux faces ont des propriétés visuelles souvent très différentes. La plupart du temps, la face orientée vers la lumière (face supérieure) est plus sombre et un peu plus spéculaire que l'autre face (face inférieure). On doit donc avoir recours à deux textures et non une seule.

Le processus pour obtenir deux textures coïncidant parfaitement d'une face à l'autre est décrit dans la Figure 75, suivie d'un descriptif détaillé de chaque étape.

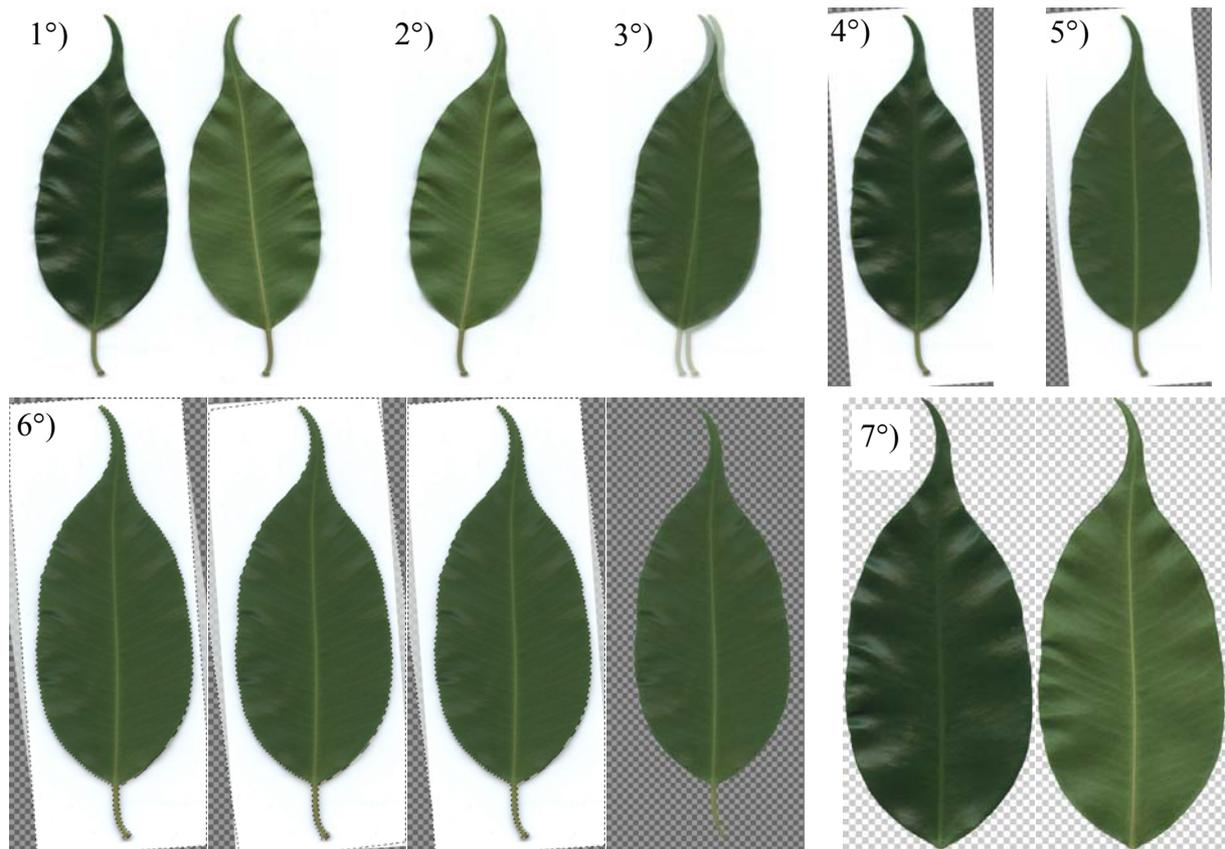


Figure 75 : Etapes d'acquisition de chacune des deux textures des faces d'une feuille. Les pointillés de l'étape 6 délimitent la zone sélectionnée.

- Etape 1 : Numériser chaque face de la feuille (supérieure et inférieure) avec la même résolution.
- Etape 2 : Inverser horizontalement la face inférieure. Ceci sert d'abord pour l'étape suivante mais permettra également au final d'affecter les coordonnées de textures aux sommets du modèle polygonal de la feuille en une seule étape.
- Etape 3 : Superposer les deux images (sur 2 couches différentes) en donnant une opacité de 50% à celle du dessus pour pouvoir voir les deux couches à la fois.
- Etape 4 : Faire pivoter la face supérieure pour bien l'aligner avec l'axe des ordonnées. Note : les damiers représentent les zones transparentes.
- Etape 5 : Ajuster (pivoter et déplacer) la face inférieure pour qu'elle soit parfaitement superposée sur la face supérieure en profitant de l'opacité à 50% pour voir en temps réel si les deux feuilles sont bien superposées.

- Etape 6 : Sélectionner le fond de chaque face avec une certaine tolérance pour englober les petites poussières du fond et ne pas laisser trop d'ombres autour de la feuille. Puis prendre l'union de ces deux sélections et effacer (remplacer par de la transparence) tout ce qui a été sélectionné. De cette façon il ne devrait rester aucune zone car les parties de la feuille qui créent une ombre d'un côté ne la recréent pas lorsqu'on les retourne.
- Etape 7 : Si l'on souhaite représenter le pétiole volumineux à l'aide d'un segment de branche, alors il faut effacer le pétiole de la texture par quelques coups de gomme sur chacune des couches ; sinon on peut le conserver s'il est naturellement assez plat ou fin. Puis découper les bords inutiles de la texture pour limiter l'utilisation de la mémoire et enregistrer enfin chaque couche (texture) indépendamment.

Le processus décomposé ci-dessus est très simple et peut être accompli en moins d'une minute avec un peu d'habitude. De plus, il est possible d'automatiser une grande partie des étapes, à l'exception les étapes 4 et 7 et éventuellement la 5.

4.3 Disposition des feuilles sur l'arbre

4.3.1 Instanciation, orientation et positionnement

Une fois créées, les feuilles sont placées à l'extrémité de chaque segment de branche terminal du squelette de l'arbre. Ces segments serviront donc de pétiole ou de rameau et pourront être texturés avec une texture différente du reste de l'arbre. La face « supérieure » (cf. paragraphe 1.2.4.2) des feuilles doit être orientée vers le haut, avec une certaine variation aléatoire, afin de respecter le phénomène naturel d'exposition des feuilles à la lumière pour optimiser la photosynthèse.

Toutefois, en considérant le très grand nombre de feuilles (plusieurs milliers) que supporte un arbre et la forte ressemblance des feuilles entre elles, il est impensable et inutile de faire en sorte que chaque feuille soit unique. La solution couramment retenue (comme par exemple par Viennot et al. [127]) est de modéliser un panel de quelques feuilles et de les « instancier » sur l'arbre ; c'est-à-dire créer de multiples instances d'une même feuilles afin de réduire considérablement les coûts en mémoire en ne définissant

qu'une seule fois chaque modèle de feuille et en n'ayant finalement pour chaque instance que des informations de position, d'orientation et éventuellement d'homothétie.

Ainsi, le positionnement et l'orientation de chaque feuille se fera de la façon suivante. Soit R le repère global de l'arbre, d'origine O , défini par les vecteurs unitaires orthogonaux $(\vec{i}, \vec{j}, \vec{k})$, tel que l'arbre part de l'origine et pousse dans la direction \vec{k} . Soit R_b $(\vec{x}_b, \vec{y}_b, \vec{z}_b)$ d'origine O_b le repère de chaque segment de branche terminal qui portera une feuille de façon à ce que la base du segment soit à l'origine et que son axe soit aligné avec un des vecteurs du repère (par exemple \vec{z}_b). Et soit R_f $(\vec{x}_f, \vec{y}_f, \vec{z}_f)$ le repère d'origine O_f dans lequel aura été modélisée chaque feuille de façon à ce que la base de son limbe soit à l'origine du repère et qu'elle soit définie principalement dans le plan (\vec{x}_f, \vec{y}_f) en ayant son extrémité dans la direction \vec{y}_f . La Figure 76 résume ces différents repères.

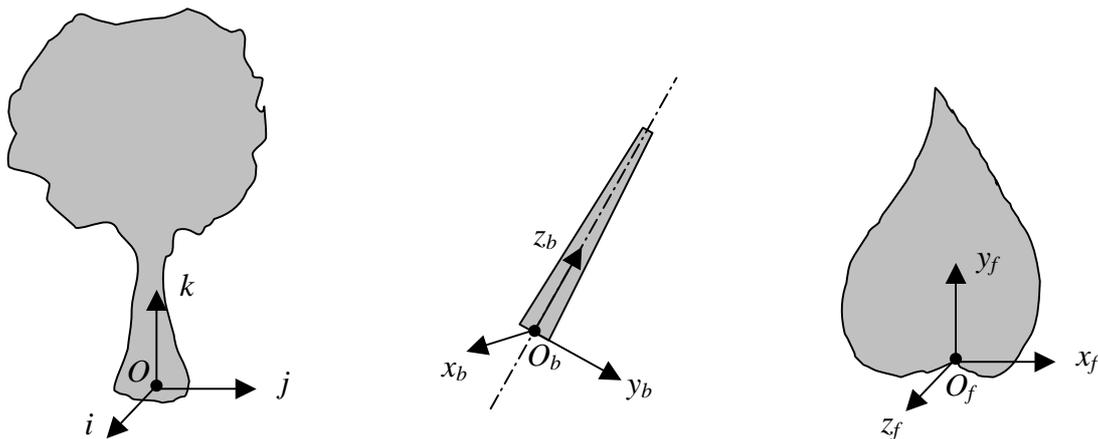


Figure 76 : Les différents repères utilisés pour les branches et les feuilles de l'arbre. De gauche à droite : arbre, segment de branche terminal (pétiole ou rameau), feuille.

On orientera alors chaque feuille en deux temps. La première étape alignera la feuille avec l'axe du pétiole tandis que la seconde étape orientera la feuille face à la lumière :

- Etape 1 : aligner \vec{y}_f avec \vec{z}_b par rotation d'angle $\widehat{\vec{z}_b \vec{y}_f}$ autour de l'axe orienté par $\vec{z}_b \wedge \vec{y}_f$. Les autres vecteurs du repère R_f subissent la même transformation pour donner le repère R_f' $(\vec{x}_f', \vec{z}_b, \vec{z}_f')$.

- Etape 2 : projeter \vec{k} sur le plan (\vec{x}_f', \vec{z}_f') , ce qui donne \vec{k}' et faire pivoter \vec{x}_f' et \vec{z}_f' d'un angle $\widehat{\vec{k}'\vec{x}_f'} \pm \varepsilon$ (ε étant une variation aléatoire de l'angle afin de ne pas avoir des feuilles trop « figées ») autour de l'axe $\vec{k}' \wedge \vec{x}_f'$.

Une fois que la feuille aura été orientée correctement, on pourra alors la translater de O_f vers la seconde extrémité du segment, exprimée dans le repère global R .

4.3.2 Gestion des intersections de feuilles

Un problème peut cependant se poser et il devient gênant lorsqu'on se rapproche très près des feuilles : c'est que celles-ci ont une forte probabilité de s'entrecouper ou de traverser des branches de par leur grand nombre et leur proximité.

La solution la plus simple est de faire varier l'angle de rotation de la feuille autour de l'axe de la branche, et éventuellement l'échelle de la feuille si aucune possibilité n'a été trouvée, jusqu'à ce que la feuille n'entre plus en collision avec une autre feuille ou une branche. Notons au passage qu'il est préférable de faire une approximation par excès du volume des branches plutôt que d'effectuer les tests par rapport au volume exact des branches incluant la texture d'écorce et les jonctions de branches adoucies. Cependant, c'est tout de même un traitement relativement lourd : de complexité $O(n \log n)$, n étant le nombre de feuilles.

Une autre solution, plus réaliste physiquement, pourrait être de plier la feuille pour qu'elle ne touche pas ses voisines. Mais on retrouvera la même complexité de traitement que précédemment pour un traitement beaucoup plus compliqué. De plus, cette méthode de déformation des feuilles est incompatible avec la technique d'instanciation des feuilles employée afin d'économiser la mémoire. Mais, de toutes façons, les apports de cette méthode, par rapport à la première méthode proposée qui est plus simple, seraient infimes. On préférera la donc simplicité.

4.4 Rendu des feuilles

4.4.1 Affichage 3D accéléré matériellement

Les feuilles étant représentées à base de polygones, elles peuvent être affichées en temps réel grâce aux cartes accélératrices 3D actuelles. Nous ne nous attarderons cependant pas sur ce domaine car, comme nous l'avons vu aux chapitres précédents, l'écorce et principalement les jonctions adoucies des branches ne sont pas encore adaptées à cette technique de rendu. De plus, les méthodes d'illumination globale que nous évoquerons plus loin pour la translucidité des feuilles ne s'y prêtent pas non plus.

D'une façon générale, l'affichage de polygones texturés, à l'aide d'OpenGL ou Direct3D, est trivial. En fait, la seule difficulté vient du fait que les textures de chacune des deux faces de la feuille sont différentes. C'est pourquoi on dédoublera les polygones de chaque feuille en ayant deux versions orientées chacune d'un côté différent de l'autre. On n'affichera alors que les polygones de la face visible, dont la normale est orientée vers la caméra.

Pour finir, il conviendra de lisser les normales des polygones en chaque sommet avec la méthode de Gouraud [46] afin de supprimer l'aspect polygonal des feuilles.

4.4.2 Lancer de rayons

Le lancer de rayons étant dépendant du moteur de rendu utilisé, on peut soit utiliser la même représentation que pour le rendu accéléré matériellement, soit adapter le moteur de rendu pour avoir deux fois moins de polygones à traiter. C'est ce deuxième point que nous allons développer ici.

Lorsqu'un rayon rencontrera une facette (triangulaire) de la feuille, il faudra donc déterminer sur quelle face de la feuille le rayon arrive. Si l'on calcule la normale à un triangle de manière à ce qu'elle pointe du côté supérieur de la feuille, un simple test de l'angle entre la direction du rayon et la normale au triangle rencontré est nécessaire (voir Figure 77). Si cet angle est supérieur à 90° , le rayon arrive du côté de la face supérieure, sinon il arrive sur la face inférieure.

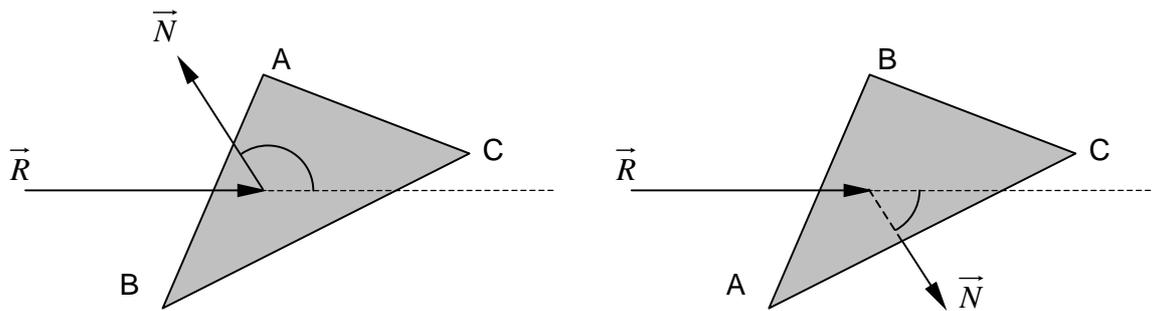


Figure 77 : Détection de la face (la normale $\vec{N} = \vec{AB} \wedge \vec{AC}$ est sur la face supérieure) visée par un rayon \vec{R} . Angle $> 90^\circ \Rightarrow$ face supérieure. Angle $< 90^\circ \Rightarrow$ face inférieure.

Si l'on associe des coordonnées (u,v) dans la (les) texture(s) de feuille à chaque coin des triangles, on peut déterminer par interpolation les coordonnées (u_i,v_i) correspondant au point d'intersection du rayon avec un triangle. Une fois que l'on sait sur quelle face on regarde, on peut donc récupérer la couleur du pixel dans la texture de la face aux coordonnées (u_i,v_i) . Pour éviter les problèmes de « pixelisation » on ne discrétisera pas les coordonnées (u_i,v_i) mais on calculera une interpolation bilinéaire entre les 4 pixels de couleur concernés.

4.5 Détection des nervures à partir d'une photo

4.5.1 Motivation

Les feuilles modélisées jusqu'ici souffrent du même problème que celui rencontré précédemment pour l'écorce : elles paraissent lisses lorsqu'on leur applique une simple texture 2D. Cette impression est certes atténuée par l'information contenue dans la texture (nervures et gondolement qui n'a pas complètement été perdu malgré l'aplatissement de la feuille sur le scanner) mais il serait néanmoins préférable de faire ressortir ce relief lors du rendu comme le montre la Figure 78.

Contrairement à la solution de déplacement de surface adoptée pour l'écorce, la finesse des feuilles nous permet de nous contenter de perturber les normales pour faire ressortir les nervures et le reste du relief de la feuille entre les nervures. La Figure 79

illustre la différence de relief entre les deux faces de la feuille. Généralement, les nervures ressortent plus nettement sur la face inférieure, tandis que la surface de la face supérieure a le relief inverse mais plus lisse ; même s'il arrive parfois que les plus grosses nervures ressortent légèrement sur la face supérieure.



Figure 78 : Comparaison entre des feuilles texturées par simple placage 2D (à gauche) et avec perturbation de normales à partir d'une carte de relief (à droite). Les zones cerclées font ressortir les principales améliorations apportées par cette seconde méthode.

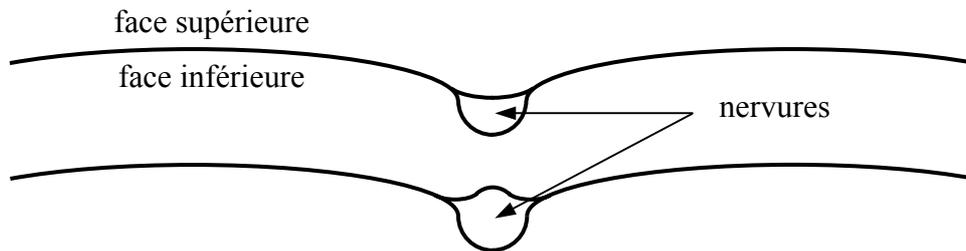


Figure 79 : Deux formes de section de nervure de feuille. Le premier cas est plus courant.

Le problème est une fois encore de pouvoir retrouver une carte du relief de la feuille qui corresponde à la photographie utilisée comme modèle. Jusqu'à présent, lorsque des nervures étaient représentées par perturbation de normales (dans des films d'animation, jeux vidéo, démonstrations techniques) elles étaient dessinées à la main. Nous avons donc présenté [81] une méthode simple mais efficace pour extraire de façon automatique le relief des deux faces de pratiquement n'importe quelle feuille à partir des textures fournies par l'utilisateur, et plus précisément à partir d'une seule d'entre elles : celles où les nervures sont les plus apparentes (c'est à dire dans la quasi-totalité des cas la face inférieure).

4.5.2 Principe de la méthode

Il s'agit là encore d'une succession d'étapes très simples qui sont cette fois-ci basées sur des traitements d'images en niveaux de gris. Comme nous l'avons mentionné au paragraphe précédent, le relief de la face supérieure correspond à l'inverse de celui de la face inférieure mais en plus lisse. Nous allons donc commencer par extraire le relief de la face inférieure pour ensuite l'inverser et le lisser. Le processus, illustré par la Figure 80 et les détails qui suivent est décrit ci-dessous.

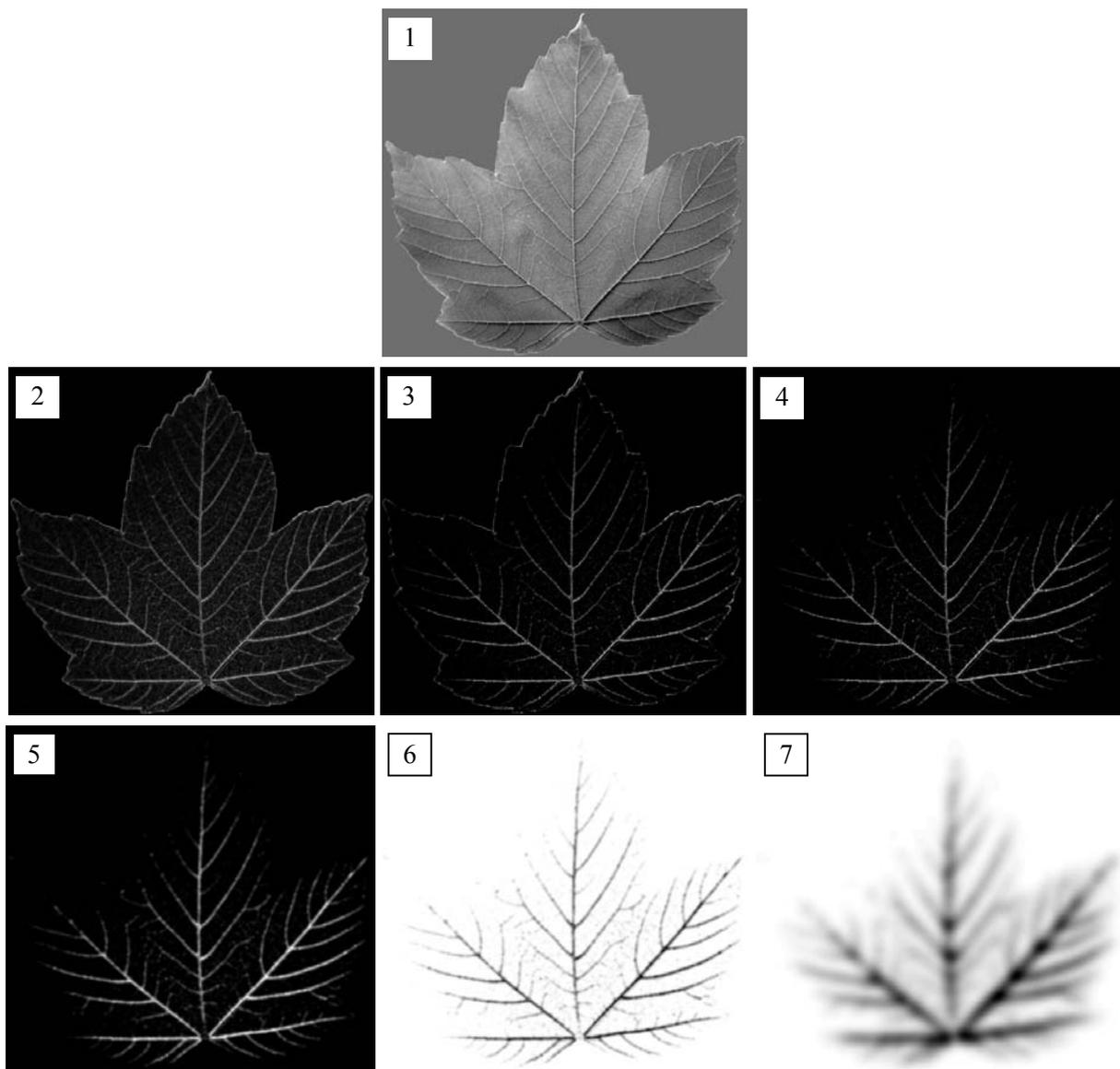


Figure 80 : Extraction du relief des deux faces d'une feuille à partir de la face inférieure.

- Etape 1 : Puisque le relief de la feuille est fortement corrélé aux nervures et que nous voulons commencer par extraire le relief de la face inférieure qui est celle où les nervures sont les plus apparentes, nous partons donc de la face inférieure passée en niveaux de gris afin de simplifier les traitements. Après avoir rempli le fond par un niveau de gris moyen, on étirera l’histogramme de l’image, afin de mieux faire ressortir les nervures. Note : pour les illustrations de la Figure 80, l’histogramme a été étiré à chaque étape afin de mieux visualiser l’effet de chaque traitement.
- Etape 2 : Effectuer un traitement d’image de détection de contours (décrit plus loin) afin de faire ressortir les nervures. Les petites nervures apparaîtront sous la forme de petits traits fins. Les nervures secondaires seront plus épaisses et plus intenses et enfin les nervures principales, encore plus larges pourront apparaître sous la forme de deux traits parallèles (chaque bord de la nervure) mais assez proches si la texture est dans une assez haute résolution, mais ceci sera corrigé à l’étape 5.
- Etape 3 : Enlever le bruit éventuel grâce à une saturation de contraste (environ 20%), ce qui a également pour but de faire ressortir les nervures principales et secondaires et d’éliminer les micro-nervures dont le relief est négligeable. Cette étape est toutefois facultative car très souvent inutile du fait du filtrage passe-bas (flou Gaussien) de l’étape 5 qui éliminera de toutes façons les pixels parasites.
- Etape 4 : Supprimer le contour de la feuille car nous ne recherchons que les nervures. Ceci peut être fait facilement en utilisant les pixels non transparents de l’image d’origine comme un masque, en érodant ce dernier de quelques pixels puis en noircissant tous les pixels hors du masque.
- Etape 5 : Appliquer un flou Gaussien (filtrage passe-bas) plus ou moins léger suivant la résolution (un petit noyau de 3x3 ou 5x5 pixels suffit pour une photo de dimensions proches de 500x500). Ceci permet d’une part d’adoucir la forme des nervures et d’autre part de relier les deux bords des nervures principales comme cela a été mentionné à l’étape 2. L’image résultante, une fois son histogramme étiré, servira de carte de hauteurs de la face inférieure.
- Etape 6 : La carte de relief de la face inférieure est réutilisée pour générer celle de la face supérieure. Commençons par l’inverser (négatif de l’image).

- Etape 7 : Appliquer un filtrage de flou Gaussien (environ 5 ou 6 fois plus fort que celui de l'étape 5). L'image qui en résulte, une fois son histogramme étiré, servira de carte de hauteurs de la face supérieure. Dans le cas où la feuille ferait partie des quelques-unes pour lesquelles les nervures ressortent légèrement sur la face supérieure, il suffit de mélanger 80% de la carte de relief obtenue avec 20% du relief de la face inférieure.

Bien qu'il existe un grand nombre de méthodes de détection de contours [52], l'algorithme que nous employons à l'étape 2 est très simple et consiste à appliquer, pour chaque pixel de l'image originale en niveaux de gris, la somme (divisée par 8, même si c'est facultatif car l'histogramme de l'image résultante sera ensuite étiré) des différences (en valeurs absolues) entre ce pixel et ses huit voisins directs. C'est-à-dire, si I est l'image d'origine et C l'image de contours résultante :

$$(66) \quad C(x, y) = \frac{1}{8} \sum_{i=-1}^1 \sum_{j=-1}^1 (I(x, y) - I(x+i, y+j))$$

La Figure 81 ci-dessous illustre deux autres exemples de feuilles. Notons que dans le deuxième exemple, les petites tâches sombres sur la feuille sont détectées par le traitement de détection de contours et génèrent des petites déformations du relief particulièrement visibles. Ce phénomène est amplifié par le fait que les nervures de la feuille sont très peu contrastées. Il faudra donc faire attention à prendre un modèle sans tâches, ou bien à les retirer manuellement à l'aide d'un logiciel de retouche d'images, ou bien d'adapter l'algorithme de détection de contours pour qu'il rejette ce genre de formes qui ne ressemblent pas à des nervures.



Figure 81 : Deux autres exemples de feuilles dont nous avons extrait le relief des deux faces à partir de la face inférieure.

4.6 Translucidité et réflectance

Un point important dans le réalisme des feuilles est leur translucidité et plus généralement la façon dont la lumière traverse les feuilles (transmission définie par une BTDF²) ou se réfléchit sur elles (réflexion définie par une BRDF³). C'est le domaine de l'illumination globale sur lequel travaille, entre autres, Baranoski [5] [6]. Les résultats qu'il obtient (voir Figure 34) sont très bons et ne rencontrent que deux problèmes :

- D'une part, les calculs d'illumination globale sont lourds et le rendu d'un arbre complet avec ses milliers de feuilles serait très long. Une solution hybride adaptative pourrait être envisagée afin de n'utiliser cette méthode de rendu que sur les feuilles les plus proches.
- Baranoski suppose que les propriétés optiques de la feuille sont uniformes sur toute la surface de la feuille. Or, ce n'est pas tout à fait le cas, principalement à cause des nervures qui sont plus ou moins translucides suivant leur grosseur, comme on peut le voir sur la Figure 82.



Figure 82 : Photographies en gros plan de nervures de feuilles vues à contre jour.

En fait, pour ce second problème, on constate sur les photographies de la Figure 82 que les nervures principales, aux endroits où elles sont les plus larges, sont plus opaques que le limbe. En outre, les nervures secondaires et les petits vaisseaux laissent mieux passer la lumière que le reste du limbe. On le constate par exemple sur la deuxième photo de feuille en calculant la différence de luminosité entre un pixel de nervure et un pixel de

² Bidirectional Transmittance Distribution Function : Fonction de distribution de transmittivité bidirectionnelle.

³ Bidirectional Reflectance Distribution Function : Fonction de distribution de réflectivité bidirectionnelle.

limbe voisin, d'une part dans la zone ombrée par le doigt qui bloque la lumière et d'autre part dans une zone éclairée à contre jour. On note alors que la différence de luminosité nervure-limbe est plus de deux fois plus grande dans les zones éclairées, ce qui prouve que les petites nervures laissent passer plus de lumière, même si, à la base, elles sont plus claires que le limbe.

Il nous paraît donc envisageable de réutiliser la carte de relief de la face inférieure que nous avons extraite au paragraphe précédent pour contrôler le degré de translucidité en chaque pixel de la texture de la feuille. Cependant, il faudrait alors ne pas supprimer les petits vaisseaux en supprimant l'étape 3 de l'extraction de relief. En fait, cette carte pourrait difficilement être utilisée telle-quelle car il est plus facile d'en déduire une hauteur des nervures qu'une largeur. Mais si l'on considère que les nervures sont plus ou moins cylindriques, la hauteur et la largeur des nervures sont liées, en théorie. C'est en tout cas une piste à étudier.

4.7 Conclusion

Nous avons présenté une panoplie d'outils extrêmement simples à mettre en œuvre visant à modéliser puis à visualiser des feuilles photo-réalistes. Ils permettent en effet de modéliser des feuilles aux contours correspondant parfaitement à de vraies feuilles et dont le relief des nervures et des bosses est pris en compte. Les résultats s'en trouvent grandement améliorés par rapport à un simple placage de textures et ceci de façon presque totalement automatisable.

Couplés à une méthode d'illumination globale telle que celle résultant des travaux de Baranoski, il serait alors possible d'obtenir des images synthèse de feuilles très réalistes. Nous avons alors indiqué les deux principaux défaut qu'il reste à corriger (la lenteur et l'absence de gestion des nervures) et avons ouvert quelques pistes à explorer dans des travaux futurs.

Chapitre 5

Résultats

Ce chapitre illustre quelques images résultant de nos travaux. La Figure 83 est un clin d'œil au « Puissant Erable » de Bloomenthal [12] et les figures suivantes (Figure 84 et Figure 85) présentent quelques arbres virtuels comportant de l'écorce et des feuilles provenant d'espèces d'arbres différentes. Le rendu de chacune de ces images, calculées en haute résolution (1024x768) avec anti-crénelage par sur-échantillonnage (3x3) sur un PC avec processeur Intel Pentium4 à 2,4 GHz, a pris un peu moins de quatre heures. La partie la plus lente du rendu concerne l'adoucissement des jonctions de branches (cf. 3.2.2).



Figure 83 : Un érable virtuel.



Figure 84 : Deux exemples d'arbres virtuels.



Figure 85 : Deux autres arbres virtuels avec des textures d'écorce peu adaptées aux méthodes que nous avons proposées. L'écorce de l'image du haut ne se prête pas au mieux à notre synthèse de texture qui introduit un effet de flou, bien qu'elle soit censée le limiter. Dans l'image du bas, la texture contient des écailles qui se chevauchent et dont la luminosité varie indépendamment de l'éclairage, ce qui est un cas que notre méthode ne gère pas très bien. Les images générées offrent néanmoins un résultat intéressant.

Conclusion

Les arbres sont omniprésents autour de nous : à la campagne bien sûr mais aussi jusqu'au cœur de nos villes. Un des buts de la Synthèse d'Images étant de représenter le monde qui nous entoure de façon informatique, les arbres sont donc une partie importante de ce domaine. De plus, leur complexité et le caractère répétitif de leurs branches en font un sujet particulièrement intéressant à traiter par ordinateur.

Les progrès significatifs qu'a connu, depuis ses débuts, le domaine de la représentation d'arbres sur ordinateur permettent aujourd'hui de visualiser des arbres de façon particulièrement réaliste. C'est cependant surtout vrai pour des arbres vus de loin. En effet, de nombreux travaux, souvent effectués en association avec des botanistes ou à partir de leurs études, permettent de modéliser de façon particulièrement réaliste (physiquement parlant) le « squelette » d'un arbre : i.e. sa morphologie, son port, ses branches, etc. En étoffant ce squelette au moyen de volumes cylindriques et de feuilles définies par un ensemble de polygones, on a pu alors obtenir des images de synthèse d'arbres visuellement réalistes à une certaine distance où l'on ne pourrait pas discerner les détails de l'arbre.

En revanche, lorsqu'il s'agit de visualiser un arbre de près, l'absence de ces détails devient visible, principalement au niveau de la texture rugueuse de l'écorce, des transitions douces entre les branches (pour ne pas reconnaître les formes primitives utilisées pour donner du volume aux branches) et, dans une moindre mesure, les nervures des feuilles. Ces trois domaines sont donc ceux que nous avons étudiés dans cette thèse et pour lesquels nous avons proposé des méthodes, simples pour l'utilisateur, lui permettant de modéliser et de visualiser ces détails lors du rendu d'un arbre vu de près en images de synthèse.

Travaux réalisés

Pour pouvoir appliquer une texture rugueuse d'écorce sur les branches, il faut d'abord être en mesure de, soit la synthétiser, soit l'acquérir. Nous avons vu que les méthodes de synthèse les plus efficaces étaient celles qui se basaient sur des photographies

d'écorce car elles permettent de tirer profit des détails de n'importe quelle écorce photographiée, tandis qu'une méthode de synthèse pure serait très difficile à concevoir, compte tenu de la grande diversité des écorces.

Toutefois, le problème des photographies est qu'elles sont en deux dimensions et ne font donc pas ressortir le relief de l'écorce. Nous avons donc étudié les méthodes permettant de retrouver le relief d'un objet et nous en sommes arrivés à la conclusion que les méthodes existantes nécessitaient un équipement coûteux et/ou encombrant et, pour les plus économiques, nécessitaient plusieurs photographies prises selon certaines conditions assez contraignantes. Afin de faciliter la tâche de l'utilisateur et de la rendre accessible à tous, nous avons donc présenté trois méthodes permettant de retrouver un relief visuellement cohérent à partir d'une seule photographie, prise dans des conditions assez peu contraignantes ; ce qui permet même de pouvoir utiliser des photographies trouvées dans des livres, des collections de textures ou sur Internet. L'ensemble de ces trois méthodes d'acquisition (extraction d'informations à partir d'une photographie) permettent de traiter une très grande majorité d'écorces.

Une fois en possession d'un échantillon d'écorce, comprenant couleur issue de la photographie et le relief qu'on en a extrait, il faut encore pouvoir l'appliquer sur les branches sans répétition ni discontinuités visibles de ce morceau de texture. Pour cela de très nombreuses méthodes ont été proposées ces dernières années pour synthétiser des textures à partir d'un échantillon. Toutes ces méthodes ont leurs avantages et leurs inconvénients (dimensions limitées sans possibilité de dupliquer l'échantillon sans discontinuités sur les bords ; effet de flou ; consommation de mémoire excessive ; temps de calcul trop longs ; etc.). Nous avons donc proposé notre propre méthode, conçue principalement sur les textures naturelles plus ou moins chaotiques comme l'écorce, en nous inspirant des avantages des autres méthodes tout en évitant leurs inconvénients afin de pouvoir synthétiser des textures « infinies » – et donc continues – à partir d'un échantillon, sans consommer trop de mémoire ni de temps de calcul lors du rendu.

L'application de cette texture indexée par des coordonnées cylindriques (u,v) sur une forme aussi complexe qu'un arbre n'est pas une chose évidente ; d'autant plus que nous souhaitons modéliser les branches de façon à pouvoir adoucir leurs jonctions tout en laissant la possibilité de « creuser » le relief de notre écorce. Nous avons adouci les

jonctions de branches en mélangeant des surfaces implicites représentant chaque segment de branche. Les méthodes existantes permettant ce mélange étaient toutes destinées à des surfaces lisses et souffraient pour la plupart d'un problème de gonflement bien connu dans le domaine du mélange de surfaces implicites. Nous avons donc proposé une méthode, spécifiquement destinée à des segments de branche modélisés par cône-sphères implicites, permettant de les mélanger sans gonflement à leurs jonctions mais permettant d'y appliquer notre texture rugueuse par déplacement de surface.

Enfin, partant du même constat que pour l'écorce, nous avons remarqué que l'utilisation de photographies de feuilles garantissait un certain réalisme visuel beaucoup plus difficile à atteindre par des méthodes de synthèse pure. Toutefois, l'utilisation de photographies posait une fois de plus le problème de l'absence de relief. C'est pourquoi nous avons proposé une méthode permettant d'extraire le relief des feuilles, issu principalement des nervures. Ces nervures peuvent alors être utilisées pour perturber la normale en chaque point de la feuille ou bien pour modifier la façon dont la lumière traverse une feuille dans le cas d'un rendu par illumination globale.

Perspectives futures

Le principal inconvénient de notre méthode de rendu des branches est la lenteur du lancer de rayons qui est nécessaire aux jonctions adoucies des branches modélisées par surfaces implicites mélangées. En effet, sans cela, une grande partie et peut-être même la totalité du rendu pourrait se faire plus ou moins en temps réel. Le déplacement de surface en serait simplifié et pourrait être fait au moyen des dernières capacités des cartes graphiques, de même que l'affichage des feuilles par polygones avec perturbation de normale pour les nervures. En ce qui concerne les ombres auto-portées de l'écorce, des travaux très récents permettent également de le faire en temps réel [130], au moyen de « *Pixel Shaders* » (fonctions de calcul de l'éclairage de chaque pixel d'une image affichée grâce aux capacités d'accélération 3D des cartes graphiques). Il serait donc très intéressant de pouvoir effectuer en temps réel le mélange des surfaces implicites représentant les segments de branche tout en conservant la possibilité de déplacer la surface pour appliquer la texture rugueuse d'écorce, ce qui reste encore l'obstacle majeur. Une piste à étudier serait de stocker un maillage « polygonisé » adaptatif de la surface lisse des branches qui

servirait de support au déplacement de surface. Mais il se pose le problème de la consommation de mémoire pour stocker tous ces polygones et surtout de la compatibilité avec notre méthode de paramétrisation de la texture d'écorce qui ne dépend pas des sommets du maillage et dont la transition d'un échantillon de texture à l'autre sur une même facette est loin d'être triviale.

D'autre part, bien que les résultats que nous obtenons apportent déjà de nettes améliorations au rendu des arbres lorsqu'ils sont vus de près, il reste encore de nombreux éléments qui participent visuellement au réalisme d'un arbre et que nous n'avons pas pris en compte. On peut citer principalement la mousse et le lichen, qui poussent en abondance sur le tronc et les branches des arbres. Une solution serait d'appliquer le lichen tout comme nous appliquons des épines au paragraphe 3.4 et de traiter la mousse comme une « *hypertexture* » [92] qui s'applique par-dessus la texture d'écorce. Mais le problème est qu'il est beaucoup plus facile de modéliser des épines que du lichen. Et l'extraction d'éléments comme de la mousse à partir d'une photographie semble très improbable car, contrairement aux écorces que nous avons pu traiter, la mousse est en 3D et non en 2D^{1/2}.

Nous avons également émis quelques idées pour améliorer la gestion de la translucidité des feuilles, sur laquelle a principalement travaillé Baranoski [5] [6], en apportant une information supplémentaire issue des nervures. Mais il reste encore beaucoup de réglages à apporter, notamment pour étudier précisément la transmission de la lumière au travers de ces nervures, selon leurs dimensions et aussi probablement l'espèce de la feuille et pour adapter la méthode à la carte de relief des nervures que nous extrayons du modèle photographique de feuille.

Enfin, tout comme nous avons différencié les deux niveaux de rendu d'arbres qui sont la visualisation de près et celle de loin, nous pourrions encore considérer un troisième niveau de rendu : celui vu de *très* près. En effet, il y a encore un grand nombre de détails que nous avons éludés et qui deviennent importants lorsqu'on observe un arbre de très près ; à commencer par les bourgeons mais aussi une forme plus précise des pétioles des feuilles, etc.

Bibliographie

Listes des articles

- [1] G. Aas, A. Riedmiller. *Les arbres feuillus*. Editions Nathan, 1994.
- [2] G. J. Agin. *Representation and description of curved objects*. Stanford Artificial Intelligence report, Memo AIM-173, 1972.
- [3] M. Aono, T. L. Kunii. *Botanical tree image generation*. IEEE Computer Graphics and Applications, 1984.
- [4] M. Ashikhmin. *Synthesizing natural textures*. ACM Symposium on Interactive 3D Graphics, 2001.
- [5] G. V. G. Baranoski, J. G. Rokne. *An algorithmic reflectance and transmittance model for plant tissue*. Computer Graphics Forum, Vol. 16, Num. 3, 1997.
- [6] G. V. G. Baranoski. *Biologically and physically-based rendering of natural scenes*. PhD dissertation, University of Calgary, 1998.
- [7] Z. Bar-Joseph, R. El-Yaniv, D. Lischinski, M. Werman, *Texture mixing and texture movie synthesis using statistical learning*. IEEE Transactions on Visualization and Computer Graphics, Vol. 7, Num. 2, pages 120–135, 2001.
- [8] B. Beneš. *Visual model of plant development with respect to influence of light*. Computer Animation and Simulation'97, 1997.
- [9] P. Bézier. *Essai de définition numérique des courbes et des surfaces expérimentales*. Thèse soutenue à l'Université Pierre et Marie Curie de Paris, 1977.
- [10] T. O. Binford. *Visual perception by computer*. IEEE Conference on Systems and Controls (Miami), 1971.

- [11] J. F. Blinn. *Simulation of wrinkled surfaces*. Computer Graphics (Siggraph'78), Vol. 12, Num. 3, pages 286–292, 1978.
- [12] J. Bloomenthal. *Modeling the Mighty Maple*. Computer Graphics (Siggraph'85), Vol. 19, Num. 3, pages 305–311, 1985.
- [13] J. Bloomenthal, B. Wyvill. *Interactive techniques for implicit modeling*. Symposium on Interactive 3D graphics, 1990.
- [14] J. Bloomenthal, K. Shoemake. *Convolution surfaces*. Computer Graphics, Vol. 25, Num. 4, 1991.
- [15] J. Bloomenthal. *Skeletal design of natural forms*. PhD dissertation, University of Calgary, 1995.
- [16] J. Bloomenthal. *Bulge elimination in convolution surfaces*. Computer Graphics Forum, Vol. 16, Num. 1, pages 31–41, 1997.
- [17] J. S. De Bonet. *Multiresolution sampling procedure for analysis and synthesis of texture images*. Computer Graphics (Siggraph'97), pages 361–368, 1997.
- [18] G. Borgefors. *Distance transformations in digital images*. Computer Vision Graphics and Image Processing, Vol. 34, pages 344–371, 1986.
- [19] J. A. Borse, D. F. McAllister. *Real-time image-based rendering for stereo views of vegetation*. Electronic Imaging'02, 2002.
- [20] J. Bouchon, P. de Reffye, D. Barthélémy. *Modélisation et simulation de l'architecture des végétaux*. INRA Editions, 1997.
- [21] G. Callen. *Les conifères*. Les pratiques du jardinage, Editions Larousse, 1989.
- [22] C. Chaudy. *Modélisation et rendu d'images réalistes de paysages naturels*. Thèse soutenue à Grenoble, 1997.
- [23] M. Chover, R. Viv, R. Quirs, X. Lluch. *Texture, displacement and immersion: a model for tree rendering*. WSCG'95, 1995.

- [24] M. F. Cohen, J. Shade, S. Hiller, O. Deussen. *Wang Tiles for image and texture generation*. ACM Transactions on Graphics, Vol. 22, Num. 3, pages 287–294, 2003.
- [25] R. L. Cook. *Shade trees*. Computer Graphics (Siggraph'84), Vol. 18, Num. 3, pages 223–231, 1984.
- [26] K. J. Dana, B. van Ginneken, S. K. Nayar, J. J. Koenderink. *Reflectance and texture of real-world surfaces*. ACM Transactions on Graphics, Vol. 18, Num. 1, pages 1–34, 1999.
- [27] X. Decoret, G. Schaufler, F. Sillion, J. Dorsey. *Multi-layered imposter for accelerated rendering*. Computer Graphics Forum (Eurographics'99), Vol. 18, Num. 3, pages 61–73, 1999.
- [28] O. Deussen, T. Strothotte. *Computer-generated pen-and-ink illustration of trees*. Computer Graphics (Siggraph'00), pages 13–18, 2000.
- [29] J.-M. Dischler, D. Ghazanfarpour, R. Freydier. *Anisotropic solid texture synthesis using orthogonal 2D views*. Computer Graphics Forum, Vol. 17, Num. 3, pages 87–95, 1998.
- [30] J.-M. Dischler, D. Ghazanfarpour. *Interactive image-based modeling of macrostructured textures*. IEEE Computer Graphics and Applications, pages 66–74, janvier–février 1999.
- [31] J.-M. Dischler. *A general model of animated shape perturbation*. Graphics Interface'99, pages 140–147, 1999.
- [32] J.-M. Dischler, K. Maritaud, D. Ghazanfarpour. *Coherent bump map recovery from a single texture image*. Graphics Interface 2002, pages 201–208, 2002.
- [33] J.-M. Dischler, K. Maritaud, B. Lévy, D. Ghazanfarpour. *Texture particles*. Computer Graphics Forum (Proceedings of Eurographics'02), Vol. 21, Num. 3, 2002.
- [34] I. Drori, D. Cohen-Or, H. Yeshurun. *Fragment-based image completion*. ACM Transactions on Graphics (SIGGRAPH 2003), Vol. 22, Num. 3, pages 303–312, 2003.

- [35] A. A. Efros, T. K. Leung. *Texture synthesis by non-parametric sampling*. IEEE International Conference on Computer Vision, 1999.
- [36] A. A. Efros, W. T. Freeman. *Image quilting for texture synthesis and transfer*. Computer Graphics (Siggraph'01), 2001.
- [37] E. Ferley, M.-P. Cani Gascuel, D. Attali. *Skeletal reconstruction of branching shapes*. Implicit Surfaces'96, 1996.
- [38] F. Di Fiore, W. Van Haevre, F. Van Reeth. *Rendering artistic and believable trees for cartoon animation*. Computer Graphics International (CGI 2003), 2003.
- [39] K. W. Fleischer, D. H. Laidlaw, B. L. Currin, A. H. Barr. *Cellular texture generation*. Computer Graphics (Siggraph'95), 1995.
- [40] B. V. Funt, M. S. Drew, M. Brockington. *Recovering shading from color images*. European Conference on Computer Vision, 1992.
- [41] G. Y. Gardner. *Simulation of natural scenes using textured quadric surfaces*. Computer Graphics (Siggraph'84), 1984.
- [42] M. Gervautz, W. Purgathofer. *A simple method for color quantization : Octree quantization*. Graphics Gems, Ed. A. S. Glassner, Academic Press, New York, pages 287–293, 1990.
- [43] J.-D. Godet, *Arbres et arbustes aux quatre saisons*. Editions Delachaux et Niestlé, 1991.
- [44] J.-D. Godet, *Guide des bourgeons de nos arbres, arbustes, arbrisseaux*. Editions Delachaux et Niestlé, 1989.
- [45] R. Gonzales, R. Woods. *Digital image processing*. Addison-Wesley Publishing, pages 173–180, 1992.
- [46] H. Gouraud. *Continuous shading of curved surfaces*. IEEE Transactions on Computers, Vol. 20. Num. 6, pages 623–628, 1971.

- [47] M. S. Hammel, P. Prusinkiewicz, B. Wyvill. *Modelling compound leaves using implicit contours*. Western Computer Graphics Symposium, 1992.
- [48] J. C. Hart. *Ray tracing implicit surfaces*. Technical Report EECS-93-014, Washington State University, 1993.
- [49] J. C. Hart. *Sphere tracing: a geometric method for the antialiased ray tracing of implicit surfaces*. The Visual Computer, Vol. 12, Num. 10, pages 527–545, 1996.
- [50] J. C. Hart. *Implicit representation of rough surfaces*. Computer Graphics Forum, Vol. 16, Num. 2, pages 91–99, 1997.
- [51] J. C. Hart, B. Baker. *Implicit modeling of tree surfaces*. Implicit Surfaces, 1996.
- [52] M. D. Heath. S. Sarkar. T. Sanocki. K. W. Bowyer, *A robust visual method for assessing the relative performance of edge-detection algorithms*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, Num. 12, pages 1338–1359, 1997.
- [53] P. S. Heckbert. *Color image quantization for frame buffer display*. Computer Graphics (Siggraph'82), Vol. 16, Num. 3, pages 297–307, 1982.
- [54] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, D. H. Salesin. *Images analogies*. Computer Graphics (Siggraph'01), 2001.
- [55] M. Holton. *Strands, gravity and botanical tree imagery*. Computer Graphics Forum, Vol. 13, Num. 1, 1994.
- [56] H. Honda. *Description of the form of trees by the parameters of the tree-like body : Effects of the branching angle and the branch length on the shape of the tree-like body*. Journal of Theoretical Biology, 1971.
- [57] B. K. P. Horn, M. J. Brooks. *Shape from shading*. MIT Press, 1989.
- [58] C. J. Humphries, J. R. Press, D. A. Sutton. *Le multiguide nature de tous les arbres d'Europe*. The Hamlin Publishing Group ltd, Bordas, 1981.

- [59] A. Jakulin. *Interactive vegetation rendering with slicing and blending*. Eurographics, 2000.
- [60] N. Janey. *Modélisation et synthèse d'images d'arbres et de bassins fluviaux associant méthodes combinatoires et plongement automatique d'arbres et cartes planaires*. Thèse soutenue à l'Université de Franche-Comté, 1992.
- [61] C. Jirasek, P. Prusinkiewicz, B. Moulia. *Integrating biomechanics into developmental plant models expressed using L-systems*. Proceedings of the 3rd Plant Biomechanics Conference, pages 615–624, 2000.
- [62] H. Jones, J. Kaandorp. *The modelling of growing natural forms*. Eurographics 2000 (Tutorial), 2000.
- [63] B. Julesz. *Textons, the elements of texture perception, and their interactions*. Nature, Vol. 290, pages 91–97, 1981.
- [64] D. Kalra, A. H. Barr. *Guaranteed ray intersections with implicit surfaces*. Computer Graphics (Siggraph'89), Vol. 23, Num. 3, 1989.
- [65] Y. Kawaguchi. *A morphological study of the form of nature*. Computer Graphics (Siggraph'82), 1982.
- [66] Y. G. Leclerc, A. F. Bobick. *The direct computation of height from shading*. IEEE Computer Vision and Pattern Recognition, 1991.
- [67] S. Lefebvre, F. Neyret. *Synthesizing bark*. 13th Eurographics Workshop on Rendering, 2002.
- [68] K. M. Lee, C. C. J. Kuo. *Shape-from-shading with a linear triangular element surface model*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, 1993.
- [69] C. H. Lee, A. Rosenfeld. *Improved methods of estimating shape from shading using the light source coordinate system*. Artificial Intelligence, Vol. 26, 1985.

- [70] L. Liang, C. Liu, Y. Xu, B. Guo, H.-Y. Shum. *Real-time texture synthesis by patch-based sampling*. Technical Report MSR-TR-2001-40, Microsoft Research, 2001.
- [71] P. Lienhardt. *Modélisation et évolution de surfaces libres*. Thèse soutenue à l'Université Louis Pasteur de Strasbourg, 1987.
- [72] P. Lienhardt, J. Françon. *Synthèse d'images de feuilles végétales*. Rapport technique R-87-1, Département d'Informatique, Université Louis Pasteur (Strasbourg), 1987.
- [73] A. Lindenmayer. *Mathematical models for cellular interaction in development*. Journal of Theoretical Biology, 1968.
- [74] A. Lindenmayer. *Developmental systems without cellular interactions: Their languages and grammars*. Journal of Theoretical Biology, Vol. 30, pages 455–484, 1971.
- [75] B. Lintermann, O. Deussen. *Interactive modeling and animation of branching botanical structures*. Computer Animation and Simulation '96, 1996.
- [76] B. Lintermann, O. Deussen. *A modeling method and user interface for creating plants*. Computer Graphics Forum, Vol. 17, Num. 1, 1998.
- [77] B. Lintermann, O. Deussen. *Interactive modeling of plants*. IEEE Computer Graphics and Applications, janvier–février 1999.
- [78] W. E. Lorensen, H. E. Cline. *Marching cubes: a high resolution 3D surface construction algorithm*. Computer Graphics (Siggraph'87), Vol. 21, Num. 4, 1987.
- [79] J. Malik, S. Belongie, J. Shi, T. Leung. *Textons, contours and regions: cue integration in image segmentation*. International Conference on Computer Vision, pages 918–925, 1999.
- [80] B. Mandelbrot. *The fractal geometry of nature*. Editions W. H. Freeman & Co., 1982.
- [81] K. Maritaud, J.-M. Dischler, D. Ghazanfarpour. *Rendu réaliste d'arbres à courte distance*. 13èmes journées AFIG, 2000.

- [82] N. Max. *Horizon mapping: shadows for bump-mapped surfaces*. The Visual Computer, Vol. 4, pages 109–117, 1988.
- [83] N. Max. *Cone-Spheres*. Computer Graphics (Siggraph'90), Vol. 24, Num. 4, 1990.
- [84] R. Měch, P. Prusinkiewicz. *Visual models of plants interacting with their environment*. Computer Graphics (Siggraph'96), 1996.
- [85] A. Meyer, F. Neyret, P. Poulin. *Interactive rendering of trees with shading and shadows*. Eurographics Workshop on Rendering, 2001.
- [86] F. Neyret, M.-P. Cani. *Pattern-based texturing revisited*. Computer Graphics (Siggraph'99), pages 235–242, 1999.
- [87] P. E. Oppenheimer. *Real-time design and animation of fractal plants and trees*. Computer Graphics (Siggraph'86), Vol. 20, Num. 4, 1986.
- [88] H. K. Pedersen. *Displacement mapping using flow fields*. Computer Graphics (Siggraph'94), Vol. 19, Num. 3, 1994.
- [89] A. P. Pentland. *Finding the illuminant direction*. Journal of the Optical Society of America, Vol. 72, pages 448–455, 1982.
- [90] A. P. Pentland. *Linear shape from shading*. International Journal of Computer Vision, 1990.
- [91] K. Perlin. *An image synthesizer*. Computer Graphics (Siggraph'85), Vol. 19, 1985.
- [92] K. Perlin, E. M. Hoffert. *Hypertexture*. Computer Graphics (Siggraph'89), Vol. 23, Num. 3, pages 253–262, 1989.
- [93] M. Pharr, C. Kolb, R. Gershbein, P. Hanrahan. *Rendering complex scenes with memory-coherent ray tracing*. Computer Graphics (Siggraph'97), pages 101–108, 1997.
- [94] B. T. Phong. *Illumination for computer generated pictures*. Communications of the ACM, Vol. 18, Num. 6, 1975.

- [95] J. Portilla, E. P. Simoncelli. *A parametric texture model based on joint statistics of complex wavelet coefficients*. International Journal of Computer Vision, Vol. 40, Num. 1, pages 49–71, 2000.
- [96] E. Praun, A. Finkelstein, H. Hoppe. *Lapped textures*. Computer Graphics (Siggraph'00), pages 465–470, 2000.
- [97] S. Premože, W. B. Thompson, P. Shirley. *Geospecific rendering of alpine terrain*. 10th Eurographics Workshop on Rendering, pages 115–126, 1999.
- [98] P. Prusinkiewicz, A. Lindenmayer, J. Hanan. *Developmental models of herbaceous plants for computer imagery purposes*. Computer Graphics (Siggraph'88), Vol. 22, Num. 4, 1988.
- [99] P. Prusinkiewicz, M. James, R. Měch. *Synthetic topiary*. Computer Graphics (Siggraph'94), 1994.
- [100] X. Qin, E. Nakamae, K. Tadamura, Y. Nagai. *Fast Photo-Realistic Rendering of Trees in Daylight*. Eurographics 2003, Vol. 22, Num. 3, 2003.
- [101] W. T. Reeves. *Particle systems – A technique for modelling a class of fuzzy objects*. Computer Graphics (Siggraph'83), Vol. 17, Num. 3, 1983.
- [102] W. T. Reeves, R. Blau. *Approximate and probabilistic for shading and rendering structured particle systems*. Computer Graphics (Siggraph'85), Vol. 19, Num. 3, 1985.
- [103] P. de Reffye, C. Edelin, J. Françon, M. Jaeger, C. Puech. *Plant models faithful to botanical structure and development*. Computer Graphics (Siggraph'88), Vol. 22, Num. 4, pages 151–158, August 1988.
- [104] I. Remolar, M. Chover, J. Ribelles, Ó. Belmonte. *View-dependent multiresolution model for foliage*. Journal of WSCG, Vol. 11, Num. 2, pages 370–378, 2003.
- [105] A. P. Rockwood. *The displacement method for implicit blending surfaces in solid models*. ACM Transactions on Graphics, Vol. 8, Num. 4, pages 279-297, 1989.

- [106] M. Rumelhart, R. Vidal. *Les arbres feuillus*. Les pratiques du jardinage, Editions Larousse, 1991.
- [107] K. Rushforth. *Photo-guide des arbres d'Europe*. Editions Delachaux et Niestlé, 2000.
- [108] H. Rushmeier, G. Taubin, A. Guézic. *Applying shape from lighting variation to bump map capture*. Rendering Techniques'97 (Proceedings of Eurographics Workshop on Rendering), 1997.
- [109] U. Shani, D. H. Ballard. *Splines as embeddings for generalized cylinders*. Computer Vision, Graphics and Image Processing, Vol. 27, pages 129–156, 1984.
- [110] A. Sherstyuk. Fast ray tracing of implicit surfaces. Computer Graphics Forum, Vol. 18, Num. 2, pages 139–148, 1999.
- [111] A. Sherstyuk, *Convolution surfaces in computer graphics*. PhD thesis, University of Monash, 1999.
- [112] I. Shlyakhter, M. Rozenoer, J. Dorsey, S. Teller. *Reconstructing 3D tree models from instrumented photographs*. IEEE Computer Graphics and Applications, pages 53–61, 2001.
- [113] A. R. Smith. *Plants, fractals, and formal languages*. Computer Graphics (Siggraph'84), Vol. 18, Num. 3, 1984.
- [114] B. Smits, P. Shirley, M. Stark. *Direct ray tracing of smoothed and displacement mapped triangles*. Technical Report UUCS-00-008, Computer Science Department, University of Utah, 2000.
- [115] C. Soler, F. Sillion, F. Blaise, P. de Reffye. *A physiological plant growth simulation engine based on accurate radiant energy transfer*. Rapport de recherche n° 4116, INRIA, février 2001.
- [116] C. Soler, M.P. Cani, A. Angelidis. *Hierarchical pattern mapping*. ACM Transactions on Graphics, Vol. 21, Num. 3, pages 673–680, 2002.

- [117] J. Stam. *Stochastic dynamics: simulating the effects of turbulence on flexible structures*. Computer Graphics Forum (Proceedings of Eurographics'97), Vol. 16, Num. 3, pages 159–164, 1997.
- [118] J. Stam. *Aperiodic texture mapping*. Technical report R046. European Research Consortium for Informatics and Mathematics (ERCIM). 1997.
- [119] N. Stolte, R. Caubet. *Comparison between different rasterization methods for implicit surfaces*. Visualization and Modeling, pages 434–447, 1995.
- [120] N. Stolte, R. Caubet. *Fast high definition discrete ray tracing implicit surfaces*. 5th DGCI (Discrete Geometry for Computer Imagery), pages 61–70, 1995.
- [121] M. Tigges, B. Wyvill. *A field interpolated texture mapping algorithm for skeletal implicit surfaces*. Computer Graphics International, pages 25–33, 1999.
- [122] D'A. Thompson. *On growth and form*. University press, Cambridge, 1952.
- [123] P.-S. Tsai, M. Shah. *Shape from shading using linear approximation*. Journal of Image and Vision Computing, Vol. 12, Num. 8, pages 487–498, 1994.
- [124] G. Turk. *Texturing surfaces using reaction-diffusion*. PhD dissertation, University of North Carolina, Chapel Hill, 1992.
- [125] G. Turk. *Texture synthesis on surfaces*. Computer Graphics (Siggraph'01), 2001.
- [126] H. Vaucher. *Guide des écorces*. Editions Delachaux et Niestlé, 1993.
- [127] X. G. Viennot, G. Eyrolles, N. Janey, D. Arquès. *Combinatorial analysis of ramified patterns and computer imagery of trees*. Computer Graphics (Siggraph'89), Vol. 23, Num. 3, 1989.
- [128] H. Wang. *Proving theorems by pattern recognition II*. Bell Systems Technical Journal, Vol. 40, pages 1–42, 1961.
- [129] H. Wang. *Games, logic, and computers*. Scientific American, pages 98–106, 1965.

- [130] L. Wang, X. Wang, X. Tong, S. Lin, S. Hu, B. Guo, H.-Y. Shum. *View-dependent displacement mapping*. Computer Graphics (Siggraph'03), 2003.
- [131] X. Wang, L. Wang, L. Liu, S. Hu, B. Guo. *Interactive modeling of tree bark*. Pacific Graphics, 2003.
- [132] J. Weber, J. Penn. *Creation and rendering of realistic trees*. Computer Graphics (Siggraph'95), 1995.
- [133] L.-Y. Wei, M. Levoy. *Fast texture synthesis using tree-structured vector quantization*. Computer Graphics (Siggraph'00), pages 479–488, 2000.
- [134] J. Woolley. *Reflectance and transmittance of light by leaves*. Plant Physiology, Vol. 47, pages 656–662, 1971.
- [135] S. P. Worley, J. C. Hart. *Hyper-rendering of hyper-textured surfaces*. Implicit Surfaces'96, pages 99–104, 1996.
- [136] Y. Xu, B. Guo, H.-Y. Shum. *Chaos Mosaic: fast and memory efficient texture synthesis*. Technical Report MSR-TR-2000-32, Microsoft Research, 2000.
- [137] S. Yamazaki, R. Sagawa, H. Kawasaki, K. Ikeuchi, M. Sakauchi. *Microfacet billboarding*. Eurographics Workshop on Rendering, pages 175–186, 2002.
- [138] L. Ying, A. Hertzmann, H. Biermann, D. Zorin. *Texture and shape synthesis on surfaces*. 12th Eurographics Workshop on Rendering 2001.
- [139] S. Zelinka, M. Garland. *Towards real-time texture synthesis with the Jump Map*. 13th Eurographics Workshop on Rendering, 2002.
- [140] R. Zhang, P.S. Tsai, J. E. Cryer, M. Shah. *Shape from Shading: a survey*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, Num. 8, pages 690–706, 1999.
- [141] Y. Zhang, Y.-H. Yang. *Multiple illuminant direction detection with application to Image Synthesis*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, Num. 8, pages 915–920, 2001.

- [142] Q. Zheng, R. Chellappa. *Estimation of illuminant direction, albedo and shape from shading*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, Num. 7, pages 680–702, 1991.
- [143] R. Zonenschein, J. Gomes, L. Velho, L. H. de Figueiredo, M. Tigges, B. Wyvill. Texturing composite deformable implicit objects. SIBGRAPI'98, pages 346–353, 1998.

Sites Internet

- [144] *Les Arbres*. <http://les.arbres.free.fr>
- [145] *Charles Poynton's Color FAQ*. <http://www.poynton.com/ColorFAQ.html>