

Rendu non-photorealiste de formes tridimensionnelles

Adrien Mondot

▶ **To cite this version:**

Adrien Mondot. Rendu non-photorealiste de formes tridimensionnelles. Synthèse d'image et réalité virtuelle [cs.GR]. 2002. inria-00598450

HAL Id: inria-00598450
https://inria.hal.science/inria-00598450

Submitted on 6 Jun 2011

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

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

DEA Imagerie, Vision et Robotique 2001/2002

Rendu non-photoréaliste de formes tridimensionnelles

RAPPORT DE DEA

Adrien Mondot



sous la direction de Nicolas Holzschuch

Laboratoire d'accueil : iMAGIS/GRAVIR -IMAG
INRIA Rhône-Alpes
ZIRST
655, avenue de l'Europe
38330 Montbonnot Saint-Martin
FRANCE

Table des matières

1 Introduction **3**

2 État de l'art **5**
2.1 Exemples de dessins traditionnels 5
2.2 Rendu non-photoréaliste 7

3 Modèle proposé **10**
3.1 Présentation du modèle 10
3.1.1 Hypothèses 10
3.1.2 Décomposition d'un dessin 11
3.1.3 Algorithme 12
3.1.3.1 Seuillage 14
3.1.3.2 Équilibrage 16
3.2 Implémentation du modèle 18
3.2.1 Entités élémentaires de dessin 18
3.2.1.1 Les traits 18
3.2.1.2 Les éléments de surface de gravure 20
3.2.2 Opérations de composition 22
3.2.3 Contrôle de l'utilisateur 22
3.2.3.1 Interface de construction des primitives 22
3.2.3.2 Précision 22
3.2.3.3 Adoucissement 25

4 Résultats **26**
4.1 Dessin au trait 26
4.2 Gravure au burin 27
4.3 Limitations 28
4.4 Améliorations 33

5 Conclusion **34**

Chapitre 1

Introduction

L'omniprésence des images synthétisées par ordinateur, dans les films, publicité ou sur Internet est désormais un fait ; et ce qui impressionne la plupart des spectateurs est la qualité photoréaliste de ces images. Les images, comme on nous l'a souvent dit valent mieux qu'un long discours, et les informations qu'elles véhiculent peuvent prendre des formes très différentes. Après avoir longtemps cherché à atteindre un niveau satisfaisant de réalisme, beaucoup de chercheurs en infographie explorent désormais des techniques de rendu qui mettraient en avant le côté expressif des images par rapport à l'exhaustivité d'une représentation plus proche de la *réalité*. Le non-photoréalisme serait donc une alternative au rendu réaliste.

Défini par ce qu'il n'est pas, le rendu non-photoréaliste réunit l'art et la science, se concentrant moins sur le procédé et plus sur la valeur informative d'une image. Souvent des techniques qui ont été longtemps employées par des artistes peuvent être appliquées à l'infographie pour souligner des attributs subtiles et/ou pour omettre des informations superflues.

La recherche en rendu non-photoréaliste peut être séparée en trois catégories principales :

- **la simulation de medias traditionels** cherche à modéliser les propriétés physiques, liées à l'utilisation de certaines techniques (par exemple le modèle d'aquarelle développé par Curtis [1]) ;
- **la création d'image assisté** a pour objectif d'aider des utilisateurs potentiellements novices, à créer des images expressives ;
- **la génération automatique d'images** tente d'imiter les techniques et les comprétences humaines dans la création artistique.

Le dessin est une technique puissante et efficace pour les représentations graphiques. Sa beauté réside dans sa simplicité, sa précision, et son expressivité. Comme tout art, il requiert des compétences : la sensibilité et la créativité sont essentielles. Le role joué par les traits de dessin est très important ; par leur forme, leur ton, leur texture et leur style, on peut traduire une infinité de significations ; c'est celà qui fait la richesse et la difficulté de ce médium. De nombreux systèmes se sont appliqués à essayer de reproduire des dessins au trait à partir d'une image donnée [17, 18, 7], mais peu ont cherché à conserver la créativté et l'expression artistique de l'utilisateur.

Notre objectif n'est donc pas de développer un programme qui générerait un dessin automatiquement à partir d'une entrée quelconque (image ou modèle 3d), mais plutôt un prototype permettant d'étudier la construction d'un dessin, donnant une grande liberté de contrôle du rendu à l'usager, tout en laissant l'ordinateur réaliser la partie technique habituellement non triviale. Nous avons continué les travaux

de Victor Ostromoukhov [14] et Frédo Durand [2] qui allaient dans ce sens, en tentant d'unifier des techniques utilisant des primitives de dessin differentes (traits et surfaces de gravure), et d'étendre l'algorithme à des formes tridimensionnelles. De la même manière que [8, 14, 17, 18] une image de base est utilisée, tant pour fournir une aide visuelle à l'utilisateur que pour servir de référence de ton, ainsi qu'un modèle tridimensionnel, afin de pouvoir expérimenter et observer ce que pourrait être un dessin ou une gravure au burin en 3D. En effet, il est difficile de savoir ne serait-ce que ce que l'on souhaiterait obtenir dans le cas de la gravure en 3D. La gravure est une technique de dessin ardue demandant beaucoup de technique que très peu de gens possèdent. Il faut plusieurs jours à un graveur expérimenté pour produire une œuvre de qualité. Dans ces conditions, on peut vraiment se demander ce que pourrait être une scène tridimensionnelle rendue dans un style de gravure en temps interactif.

Nous commencerons par une étude des travaux qui ont été conduits sur le rendu non-photoréaliste. Ensuite nous présenterons le modèle que nous avons utilisé puis étendu (chapitre 3), et enfin quels sont les résultats obtenus (chapitre 4). Nous verrons donc comment à partir des travaux antérieurs de Durand et Ostromoukhov, nous sommes arrivés à proposer un premier modèle unifié qui gère des techniques de dessin variées tant en 2D qu'en 3D. Certes il est très loin d'être parfait, mais il permet une première étude sur un domaine *a priori* difficilement appréhendable, le dessin en 3D.

Chapitre 2

État de l'art

Dans cette partie nous verrons quelques éxemples de gravures et de dessin à la pointe sèche, et nous verrons quelles ont étés les précédents travaux dans ce domaine.

2.1 Exemples de dessins traditionnels

Les deux gravures de visages présentées dans la Fig. 2.1 montrent la subtilité de la technique : les orientations des traits servent essentiellement à suggérer le volume ; les autres informations concernant la forme sont transmises par la gestion des tons. Cette gestion se fait en changeant la taille des hachures et/ou en utilisant une superposition de hachures (c'est le cas dans le cou pour les deux gravures).



FIG. 2.1 – Exemples de gravures au burin traditionelles : un timbre de poste (à gauche) et un billet de banque (à droite).

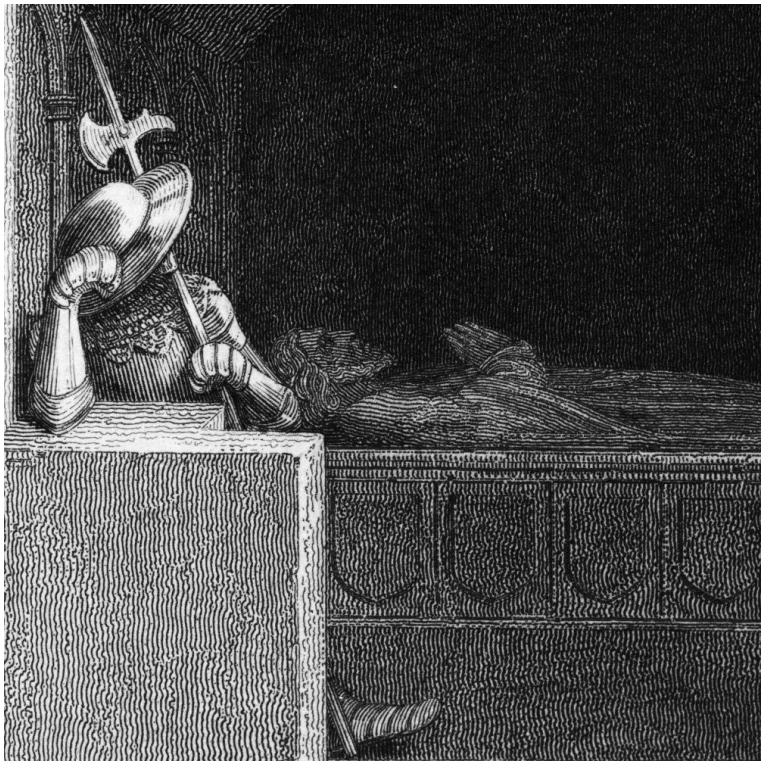


FIG. 2.2 – Exemple de gravure au burin traditionelle (détail)



FIG. 2.3 – Exemple de dessin à la pointe sèche (autoportrait de Rembrandt)

2.2 Rendu non-photoréaliste

Un certain nombre de systèmes de rendu non-photoréaliste [4, 5, 13, 16, 24, 10] construisent des images à partir d'un modèle 3D. D'autres ont choisi d'utiliser une image de référence pour spécifier les tons de l'image finale, tels que Haeberli [8].



FIG. 2.4 – Dans les travaux de Salisbery, les tons plus sombres sont obtenus en rajoutant plus de traits.

Avec une approche similaire à Haeberli, Salisbery [17, 18] génère des dessins à la plume à partir d'images en niveaux de gris (Fig. 2.4). Mais en utilisant la texture et les tons de l'image de référence, ils se limitent à un style bien spécifique de dessin, et à des traits simple. La différence entre les deux méthodes est plus importante qu'elle n'en a l'air : dans leur travail, les tons plus sombres sont obtenus en rajoutant plus de traits, alors que la nôtre nous permet d'utiliser des trais plus épais, ou plus appuyés. Cela autorise l'utilisateur à contrôler précisement le placement des traits, sans se soucier du respect du

ton, qui se fera automatiquement. Notre système, bien que beaucoup inspiré du travail de Salisbury y rajoute un contrôle de plus haut niveau, ainsi qu'un retour visuel constant durant l'édition.



FIG. 2.5 – Résultat obtenu par Hertzman [10] avec son système de hachures à la plume

En partant d'un modèle 3D, Hertzman [10] génère des dessins hachurés à la plume. Il est intéressant de remarquer que les primitives de dessin qu'il utilise ne sont pas directement attachées au modèle : il utilise une approximation de la courbure de la surface pour déterminer l'orientation des traits de hachures. Puis suivant la valeur de l'illumination locale, il détermine quelle est l'intensité de hachure à utiliser. Les traits de hachure sont générés dynamiquement, en fonction d'un certain nombre de paramètres, lors du rendu. Cependant son modèle de hachure est coûteux en temps de calcul, et l'utilisateur ne dispose que d'un contrôle très réduit sur l'image correspondante.

Webb et *al.* [23] ont développé un sytème de hachures en temps réel de modèle 3D, à base de seuillage. Afin d'accélérer le processus de création des hachures, ils utilisent des textures de hachures pré-calculées. Pour obtenir les tons désirés, ils se servent des textures comme structure de seuillage. L'inconvénient de leur méthode est que les textures de hachures sont collées directement sur la surface du modèle 3D, ce qui produit des résultats peu naturels pour des modèles complexes.

Des simulations complexes de l'interaction papier-stylo ont été proposées par différents auteurs [21, 20, 19], avec des résultats très convaincants, mais cette approche ne permet pas pour l'instant l'utilisation de brosses variées.

Le modèle de trait que nous utilisons est plus simple à généraliser (de nouveaux types de traits peuvent être créés simplement avec un scanner), et convient mieux à une approche semi automatique de modélisation de ton.

Notre simulation des traits est inspirée par la recherche sur la gravure digitale [12, 14], et tout particulièrement par les travaux d'Ostromoukhov [14] et Durand [2] que nous avons continués, où les traits de gravure sont simulés en utilisant une méthode de rendu en demi-ton : le seuillage. Notre modèle de trait peut être utilisé avec un grand nombre de médias, et dans un système temps réel.

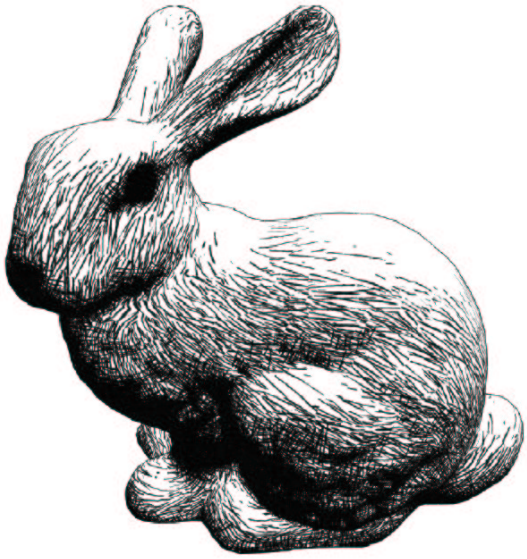


FIG. 2.6 – Résultat obtenu par Webb [23] avec son système de hachures basé sur une technique de seuillage

Nous avons développé une extension pour la 3D à l'algorithme proposé dans [2], et nous l'avons étendu pour qu'il puisse gérer d'autres primitives de dessin que les traits, tels que les patchs de gravure, comme présenté dans [14]. Comme dans [10], les primitives de dessin ne sont pas directement liées au modèle 3D. De plus notre système fonctionne en temps interactif, et l'utilisateur est le seul maître des choix esthétiques, seule la réalisation technique du dessin restant à la charge de l'ordinateur.

Chapitre 3

Modèle proposé

Le modèle que nous proposons est une extension de celui développé par Durand et al. dans [2], il se base sur une décomposition des caractéristiques constitutives d'un dessin et sur une différentiation des attributs de haut niveau — comme la précision — par rapport à leur réalisation technique — le bas niveau — les auteurs introduisent également l'idée de percevoir la génération d'un dessin comme étant un mapping multidimensionnel entre *l'espace de contrôle de l'utilisateur* et *l'espace image*. Ce système ne traitait que les traits comme primitives de dessin, nous l'avons étendu aux éléments de surface (pour la gravure au burin), et une notion de *dessin tridimensionnel* a également été développée. Dans ce chapitre, nous détaillerons dans un premier temps quel est le modèle que nous avons mis au point, puis comment il a été implémenté.

3.1 Présentation du modèle

D'après quelques hypothèses de base sur les différents aspects entrant en jeu dans la création d'un dessin, nous avons développé un modèle qui tente de prendre en compte chacune de ces composantes. Les deux premieres sections (3.1.1 et 3.1.2) présentent les hypothèses et analyses faites par Ostromoukhov et Durand. Dans la troisième section on verra comment a été adapté le modèle pour nos besoins.

3.1.1 Hypothèses

Inspiré par différents auteurs [3, 6], ils distinguent cinq composantes techniques dans le dessin figuratif :

- la composition,
- la perspective,
- les contours,
- la gestion des tons,
- le contrôle des détails.

Pour chaque composante, il est important de faire une distinction entre les choix esthétiques et la réalisation technique de l'œuvre. Il est aussi crucial de comprendre quelles sont les similitudes et les différences entre le dessin et la photographie, pour chacune de ces composantes, car le système utilise comme entrée une image de référence fournie par l'utilisateur, typiquement une photographie.

- **La composition** est un parti pris purement esthétique, et c'est pour cette raison que notre système a laissé cet aspect de côté. Il repose complètement sur l'image de référence prise comme support par l'utilisateur.
- **La perspective** peut être définie par le choix du point et du champ de vue, ainsi que par l'utilisation de la technique de projection géométrique développée depuis la renaissance. Il s'agit d'un aspect purement technique ; et la photographie utilisant généralement la même perspective, il a été choisi de conserver celle de l'image de référence.
- **Le contour** est une des composantes fondamentales du dessin au trait. Il se rapporte à la vision de bas niveau. Un outil de détection de contour a donc été incorporé, bien que ce procédé uniquement technique ne soit pas aussi efficace que le travail esthétique que pourrait réaliser un artiste.
- **La gestion des tons** a pour but de révéler les formes et les textures, de définir l'ambiance et l'équilibre de l'image ; elle peut se faire de manière systématique si l'on doit reproduire des tons *objectifs* (dans le sens clairement définis). Il s'agit de respecter ceux de l'image d'origine, l'utilisateur dispose cependant d'un outil lui permettant d'influencer localement la valeur tonale afin qu'elle corresponde mieux à ses attentes.
- **Le contrôle des détails** permet au dessinateur de souligner des zones importantes en augmentant la précision de la reproduction tonale. Faisant partie des paramètres subjectifs, l'utilisateur a la possibilité de fixer localement cette quantité de détail, *via* l'édition d'une *carte de précision*.

3.1.2 Décomposition d'un dessin

Lors de la création d'un dessin les cinq caractéristiques entrent en jeu, d'où la difficulté qu'éprouvent les débutants à gérer simultanément autant de paramètres interdépendants. Le système tente de séparer ces aspects et de fournir tant des outils d'édition de haut niveau, que des contrôles sur l'édition des traits de bas niveau. Pour réaliser ceci nous considérons (comme dans [2]) la génération d'un dessin comme un mapping multidimensionnel entre l'espace *utilisateur* et l'espace *image*.

Au centre de cette analyse se trouve la distinction entre la simulation du média et le contrôle de l'utilisateur. La simulation du média est une imitation « mécanique »de l'interaction qui a lieu entre l'outil de dessin et le support du dessin pour des techniques traditionnelles. Cette simulation est déterminée par un certain nombre de paramètres qui sont contrôlés indirectement par l'utilisateur ; c'est *l'espace de contrôle de l'image*.

L'espace de contrôle de l'image peut être considéré comme un ensemble de primitives de dessin avec des attributs, des lignes de contours, un point de visualisation, etc. Dans notre système à base de seuillage, cet espace est modélisé par une matrice de seuillage générée en fonction des primitives utilisées et du point de vue et d'une image cible donnant le ton à respecter qui est également une fonction du point de vue. La génération de l'image finale consiste à comparer la matrice de seuillage avec le ton cible afin d'obtenir le niveau de gris désiré pour chaque pixel.

L'espace *utilisateur* décrit les degrés de liberté de l'utilisateur, c'est-à-dire l'ensemble des paramètres qui vont modifier globalement ou localement le rendu. Certains de ces contrôles se font spatialement, par exemple pour déterminer la précision du rendu de certaines zones, une carte de précision est prise en compte.

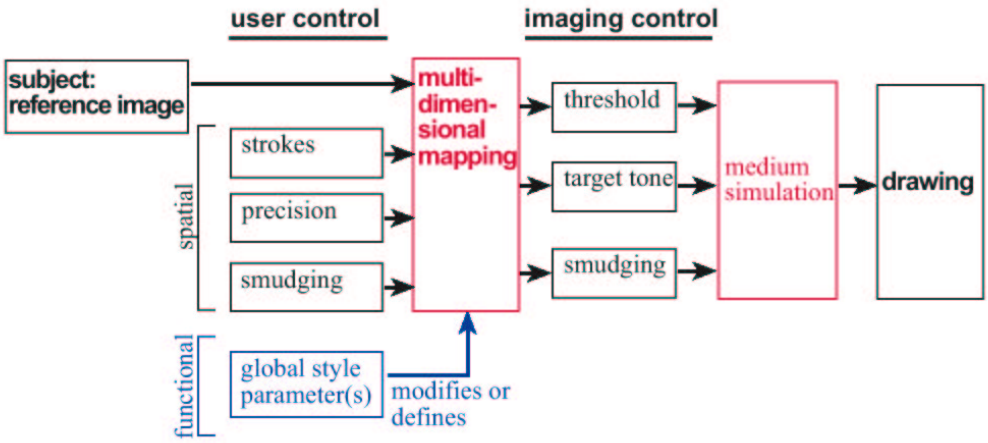


FIG. 3.1 – La génération d'un dessin défini comme un mapping entre l'espace utilisateur et l'espace image.

3.1.3 Algorithme

L'algorithme de rendu peut être considéré comme un mapping multidimensionnel entre l'espace *utilisateur* et l'*espace de contrôle de l'image*, suivi par un rendu final.

En entrée de l'algorithme on dispose :
- d'une image de référence ;
- des données tridimensionnelles associées à l'image de référence ;
- d'un point de vue.

À partir de l'image de référence, l'utilisateur définit
- les primitives de dessin, qui peuvent être des traits ou des surfaces ;
- les zones nécessitant une certaine précision dans le traitement. L'ensemble de ces données, regroupées dans une image plaquée sur le modèle 3D est appelée *carte de précision*.
- les zones qui doivent être adoucies, qui forment la *carte d'adoucissement*.

Des données intermédiaires sont calculées :
- la matrice de seuillage *brute* est générée à partir des primitives de dessin.
- la carte de couverture, indiquant pour chaque pixel de l'image le nombre de primitives se superposant en ce pixel.
- la carte des tons cible qui est une version filtrée de l'image initiale, où la quantité de détail a été spécifiée par la carte de précision.
- la carte de seuillage équilibrée : afin de compenser les histogrammes non plats résultant du placement irrégulier des traits, il est nécessaire de corriger la carte de seuillage. Cette opération nommée équilibrage se fait en fonction de la carte de précision ainsi que du nombre de primitives se recouvrant (la carte de couverture).

L'image finale est le résultat d'une opération de seuillage, qui consiste à comparer pixel par pixel la carte de seuillage équilibrée et la carte des tons cibles afin d'en déduire une valeur (un niveau de gris). Cette opération est influencée par la carte d'adoucissement, qui permet d'atténuer le seuillage.

Toutes les données de l'espace utilisateur sont définies dans un espace tridimensionnel :
– les primitives de dessin reposent sur une surface 3D sur laquelle est plaquée l'image de référence ;
– pour chaque rendu les primitives sont projetées dans l'espace image afin de générer la carte de seuillage brute et la carte de couverture ;
– de la même manière, les cartes de précision et d'adoucissement sont plaquées sur une surface, puis projetées dans l'espace image.
Le reste de l'algorithme se fait donc en deux dimensions, dans l'espace image.

Cet algorithme peut être résumé par la figure 3.2.

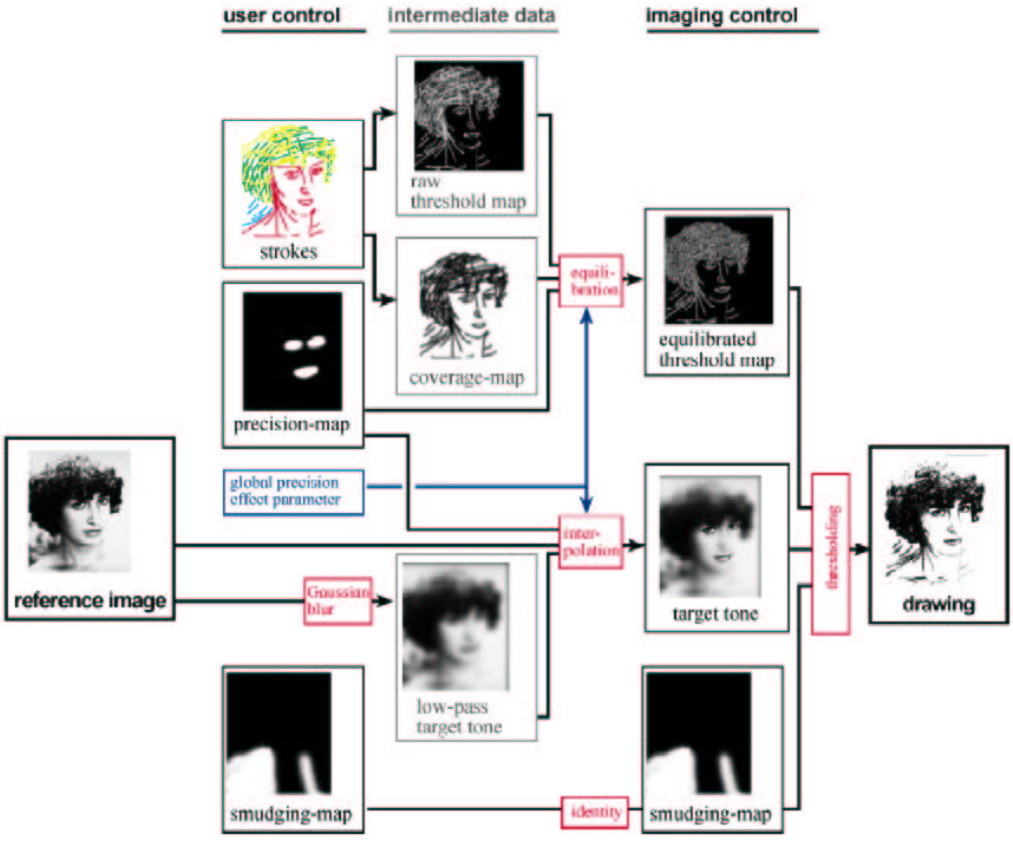


FIG. 3.2 – Le rendu du dessin est exprimé comme un ensemble de mappings 3D.

Le rendu final est une image bitmap, mais les structures de données servant à la génération de l'image sont l'ensemble des traits, les cartes de tons (image initiale) et de précision qui sont des données géométriques. À l'opposé de systèmes purement basés sur une image pixel (tel que Corel Painter), la résolution de l'image finale n'est pas fixée, car le dessin est constamment recalculé à partir des tracés vectoriels des traits. On peut donc effectuer un dessin indépendemment de la résolution de l'image d'origine.

3.1.3.1 Seuillage

Dans ce paragraphe, nous allons présenter la méthode de seuillage, qui permet d'obtenir des rendus correspondant à des techniques très différentes telles que le dessin au crayon ou la gravure au burin en respectant une intensité de ton donnée, nous utilisons la même méthode que [2, 14].

Pour le rendu en demi-ton, le ton est obtenu en comparant pour chaque pixel la valeur d'une matrice de seuillage $T(x,y)$ au ton cible $I(x,y)$ de l'image de base. En fonction du résultat, la valeur du pixel est déterminée : noir ou blanc [22]. Si S est est une fonction par seuil ($S(x) = 0$ si $x < 0$ et $S(x) = 1$ si $x \geq 0$), alors le seuillage peut être exprimé de la manière suivante :

$$Seuil(T(x,y), I(x,y)) = S(T(x,y) - I(x,y)) \tag{1}$$

Si l'histogramme de la matrice de seuillage est plat (tous les tons sont présents en même quantité), alors une reproduction fidèle est obtenue.

Dans notre système, les tracés de traits sont simulés en utilisant une structure de seuillage locale que l'on pourrait appeler une *texture de seuillage*. Pour rendre le dessin, on rend tous les tracés de traits en utilisant leurs structures de seuillage, on obtient alors une matrice globale de seuillage $T(x,y)$, qui est comparée aux tons de référence $I(x,y)$ pour obtenir un dessin en noir et blanc. C'est dans cette technique que réside la solution du problème de séparation du ton du trait de son style. La Figure 3.3 montre comment on peut obtenir différents tons à partir de la même texture de trait, la Fig. 3.4 illustre le même principe dans le cas d'un élément de surface de gravure. À l'opposé des approches précédentes aucune coupe explicite des traits n'est nécessaire, car le résultat est fourni implicitement par la méthode de seuillage.

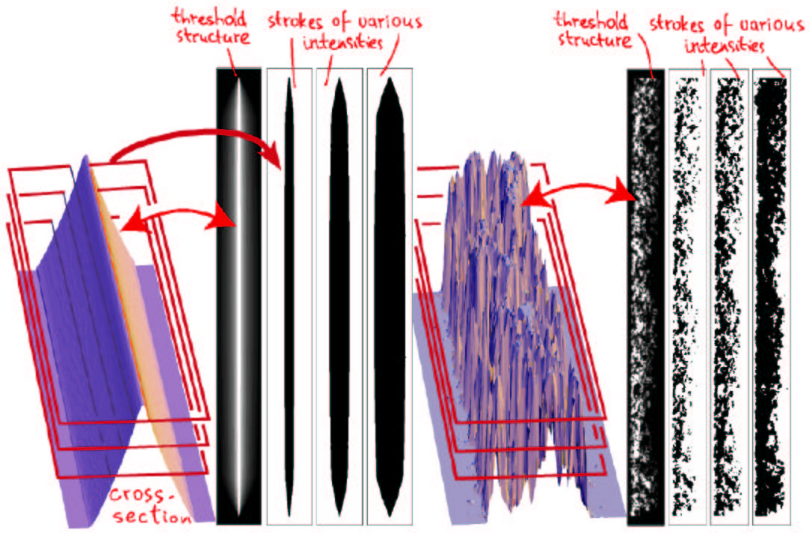


FIG. 3.3 – Structure de seuillage typique pour un trait à la pointe sèche (gauche) et de crayon (droite). On peut voir les différents effets de ton obtenus à partir d'une seule structure de seuillage. Le principe est de couper la structure à différentes hauteurs en fonction de la valeur du ton désirée (Illustration tirée de [2]).

Lorsque plusieurs traits se superposent une méthode de fusion doit être déterminée. En effet, une

simple méthode additive ou de moyenne engendre des artefacts aux jonctions des traits après le seuillage, les bords des traits paraissant plus sombre. Après expérimentation, il apparait que seul le mode maximum (*i.e.* pour deux pixels se superposant, on garde celui dont la valeur est la plus élevée) imite de manière réaliste la superposition de plusieurs traits.

Le placement irrégulier des traits entraine un changement de l'histogramme, il n'est plus plat. Il est donc nécessaire d'effectuer un rééquilibrage pour obtenir un rendu fidèle des tons ; ce rééquilibrage sera présenté dans la section suivante.

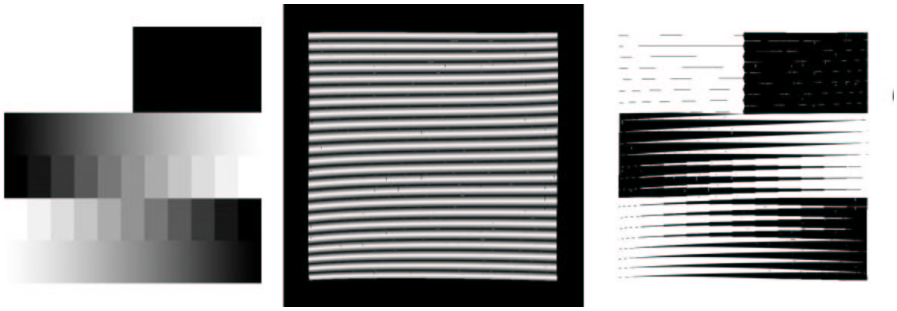


FIG. 3.4 – Exemple du seuillage dans le cas de la gravure au burin. De gauche à droite : la carte des tons cibles, la carte de seuillage et l'image résultante.

À l'origine, la technique du seuillage a été développée pour effectuer le rendu de tons sur des dispositifs haute résolution ne pouvant produire que des points noirs [22]. À l'opposé de ces dispositifs, les écrans que nous utilisons disposent d'une résolution bien moindre et d'une gamme de tons reproductibles bien plus grande. Afin d'avoir un rendu correct des images à l'écran, il est nécessaire d'introduire un dispositif d'*antialiasing* ; pour cela, on modifie la fonction S de l'équation 1 : en utilisant une fonction en forme de S on peut adoucir les tons obtenus. Cependant la conservation de la fidélité de reproduction des tons nous oblige à apporter une attention toute particulière à la symétrie de cette fonction. L'idée est d'utiliser une fonction basée sur un sinus limité à une largeur donnée $2w$. En utilisant t pour $T(x,y)$ et i pour $I(x,y)$, on a :

$$s(t,i,w) = \begin{cases} 1 & \text{si } t-i < -w \\ 0,5+0,5sin(-\frac{\pi(t-i)}{2w}) & \text{si } -w \leq t-i \leq w \\ 0 & \text{si } w < t-i \end{cases} \quad (2)$$

Ce procédé fonctionne correctement dans le cas des tons gris, mais biaise les résultats pour des tons trop clairs ou trop foncés. Ceci peut-être expliqué par le fait que l'on force dans certains cas le ton à noir ou blanc, ce qui fausse la symétrie de la fonction. Pour minimiser cet effet, on limite la largeur de la fonction s au minimum de la distance entre le ton x et le noir ou le blanc ($d(x) = min((1-x,x))$ et on utilise :

$$s'(t,i,w) = s(t,i,max(w,d(i))) \quad (3)$$

On verra dans la section 3.2.3.3 que faire varier la valeur de w permet d'adoucir localement l'image.

3.1.3.2 Équilibrage

L'équilibrage de structures de seuillage complexes en rendu en demi-tons se fait traditionellement par une approche « essai et erreur »[14, 15]. Pour chaque itération, une série de patchs de tests est rendue puis comparée au ton cible (*i.e.* le ton désiré) en prenant en compte l'intégration visuelle qu'effectue habituellement notre système visuel, la structure est alors mise à jour en correspondance. Malheureusement cette approche est évidemment trop lente pour une implémentation temps-réel. Nous utilisons donc la méthode présentée dans [2] basée sur une *look-up-table* ; ce n'est pas aussi précis que les techniques précédentes mais c'est suffisant pour nos besoins.

À partir du moment où les histogrammes des textures de seuillage de nos primitives de dessin sont plats (cf. section 3.1.3.1), on a uniquement besoin de compenser lors de la superposition de primitives. Cette approche est basée sur une simple analyse probabiliste de la valeur de l'histogramme des zones où les primitives se superposent (mélangées avec le mode *max*).

Dans le cas des traits : nous modélisons son histogramme comme étant la fonction de probabilité de densité $p(x) = 1$ pour $x \in [0,1]$. La fonction de densité cumulée (*i.e.* l'histogramme cumulé) est $p(X < x) = \int_0^x p(\xi)d\xi = x$.

Ceci est vrai si l'on suppose que les traits adjacents sont indépendants les uns des autres. Cette hypothèse reste raisonnable étant donné le placement irrégulier de nos traits et leurs textures. Les probabilités de densité et de densité cumulée de deux traits se superposant p_1 et p_2 sont :

$$P_{max(1,2)}(x) = p_1(x) * p_2(x < X) + p_2(x) * p_1(x < X) = 2x$$

$$P_{max(1,2)}(X < x) = x^2$$

Cette formule peut être généralisée à la superposition de n traits :

$$P_{max(i=1..n)}(x) = nx^{n-1}$$

$$P_{max(i=1..n)}(X < x) = x^n$$

Les mesures expérimentales que nous avons effectuées valident relativement bien cette équation (Fig. 3.5).

Un histogramme plat est alors obtenu en multipliant chaque valeur x par la valeur correspondante de l'histogramme cumulé $p_{max(i=1..n)}(X < x)$. D'après ce modèle, l'équilibrage peut être effectué en appliquant une simple fonction puissance selon le nombre de traits se superposants (voir par exemple la Fig. 3.6).

La technique d'équilibrage a dû être adaptée pour gérer la gravure au burin. En effet, le modèle probabiliste proposée dans la section 3.1.3.2 n'est pas valide avec des traits de gravure. Donc nous avons dû modifier cette méthode pour faire un équilibrage par force brute (*i.e.* : par pixel). Notre modèle consiste à conserver un histogramme pour chaque pixel de l'image, ensuite on détermine le nombre de patchs qui s'appliquent à ce point pour réduire ou augmenter l'intensité en conséquence.

On modélise l'histogramme d'un patch à l'aide de la fonction de probabilité $p(x) = 1$ pour $x \in [0,1]$. La fonction de densité cumulative (qui correspond à l'histogramme cumulé) est $P(X \leq x) = x$. Ici on suppose que les patchs n'ont aucune corrélation, ce qui peut être faux dans certains cas où deux patchs ont des fréquences ou des orientations similaires.

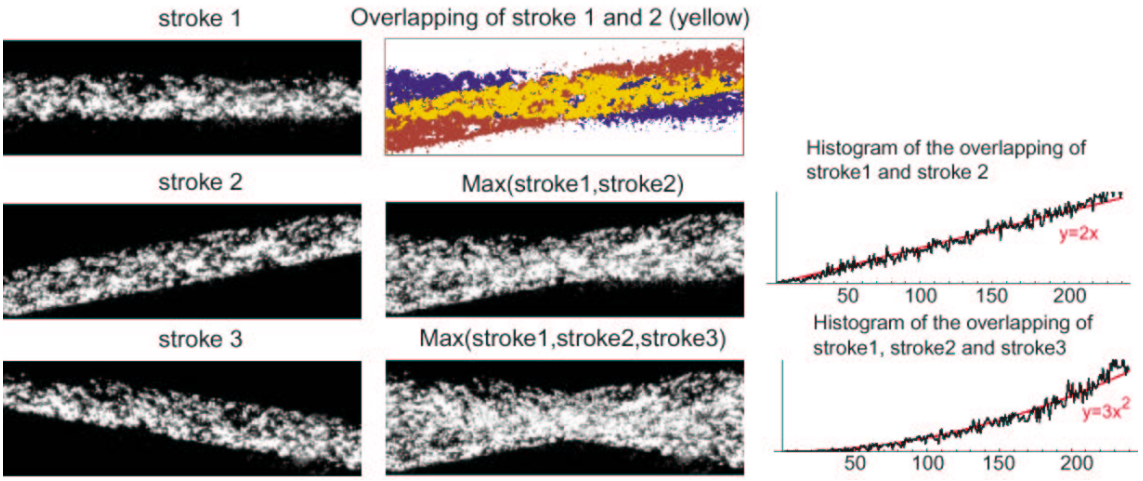


FIG. 3.5 – Histogramme de traits se superposant avec le mode *max*. La fonction obtenue théoriquement est ici en rouge

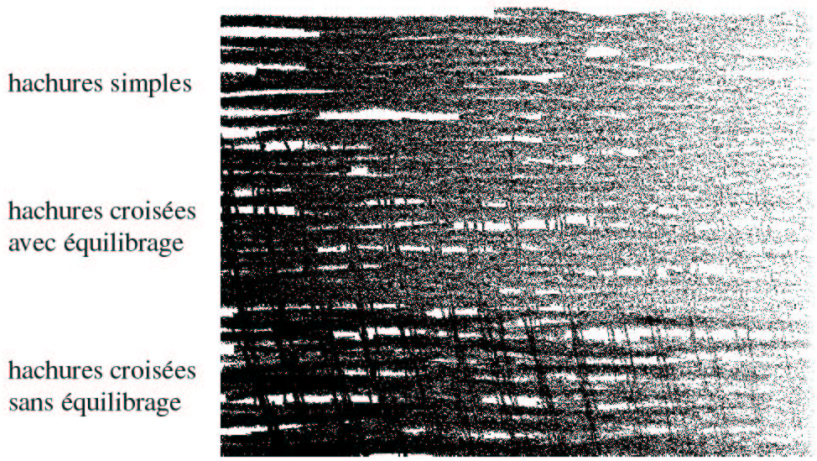


FIG. 3.6 – Effet de l'équilibrage en prenant un dégradé du noir vers le blanc comme carte de tons cibles.

Ensuite la fonction de probabilité densité pour n patchs qui se superposent en mode *max* est : $P_{max}(X \leq x) = nx^{n-1}$. Pour conserver le ton de l'image d'entrée, cette courbe doit être linéaire. Donc on élève la valeur de seuillage à la puissance n.

3.2 Implémentation du modèle

Dans la partie précédente nous avons présenté un modèle de création de dessin qui marche pour différent type de dessin, à la fois en 2D et en 3D.

Dans cette partie, nous verrons concrêtement comment est réalisé ce modèle, c'est-à-dire quelles sont les primitives de dessin, leur structure et leur rendu, puis comment ont été réalisé les méthode de seuillage et d'équilibrage (opérations de composition), et enfin quel sont les paramètres que l'utilisateur maîtrise et comment ils dirigent la création du dessin.

3.2.1 Entités élémentaires de dessin

Les entités élémentaires de dessin sont l'outil de base de création du dessin : elles sont le support de base dans la génération de la carte de seuillage.

3.2.1.1 Les traits

Les traits individuels sont décrits par un squelette et une référence vers la texture de seuillage associée au trait. Le squelette est une courbe poly-Bézier, dont les points de contrôles reposent sur une surface tridimensionnelle définie par l'utilisateur. Le squelette (défini dans l'espace objet) est projeté dans l'espace image lors de chaque rendu, puis le trait est triangulé et texturé.

Une fois que les traits sont projetés dans l'espace image, on décale légèrement leur position en z afin d'éviter les problèmes classiques de l'algorithme du Z-buffer (le manque de précision entraine des artefacts important lorsque deux surfaces sont à la même profondeur). Bien sûr la visibilité des traits n'est calculée que par rapport à l'objet sous-jacent, plusieurs traits pouvant normalement se superposer, il n'y a pas de gestion des occlusions inter-traits.

Le rendu de la matrice de seuillage se fait en habillant chaque squelette de trait par sa texture de seuillage. Il est donc nécessaire de transformer les textures de seuillage, qui sont droites, en fonction de l'orientation de la courbe de Bézier représentant le trait. On utilise une triangularisation de la bande (de taille fixe) autour de la courbe (voir Fig. 3.7) ; puis une interpolation linéaire est effectuée à l'intérieur de chaque triangle. Cette méthode est implémentée en utilisant OpenGL et le mapping de texture ; les courbes sont reparamétrisées afin de normaliser leur longueur euclidienne, et la texture est décalée aléatoirement afin d'éviter une uniformité artificielle.

La texture de seuillage est simplement une image en niveau de gris avec un histogramme plat. On peut l'obtenir à l'aide d'un scanner, ou bien la modéliser analytiquement. Nous avons numérisé de vrais traits pour le crayon et le fusain : ils ont été scannés à 1200 dpi et rééchantillonnés pour obtenir les textures de moindre résolution. Afin de respecter les contraintes pour la reproduction linéaire de ton, nous leurs avons appliqué un algorithme d'aplatissement d'histogramme (voir Fig.3.8).

Lors du rendu des traits, on utilise le mode de mélange *max* d'OpenGL 1.2 qui permet pour chaque couche de prendre le maximum entre la valeur courante et la valeur précédente.

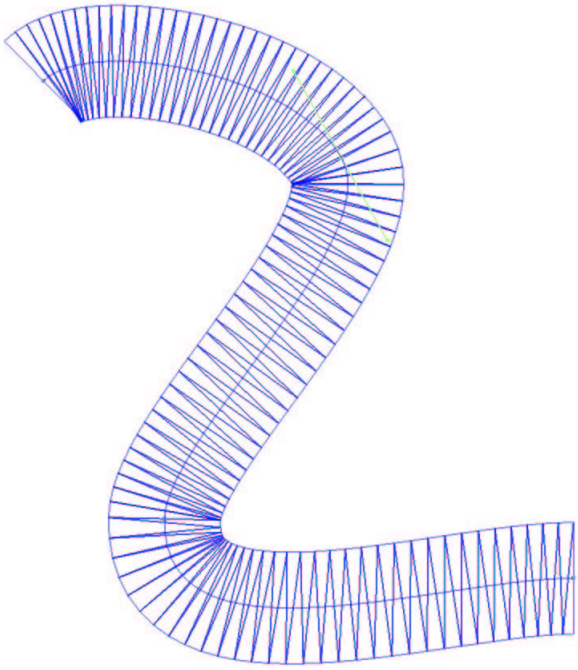


FIG. 3.7 – Structure d'un trait de dessin.



FIG. 3.8 – Textures de seuillage utilisées pour simuler des traits au crayon (à gauche) et à la pointe sèche (à droite) à différentes tailles. Pour que l'opération de seuillage reproduise de manière correcte les tons désirés, leurs histogrammes doivent être plats.

Le rendu de la carte de recouvrement se fait en remplaçant la texture de seuillage par la texture de recouvrement préfiltrée avec un filtre gaussien afin de simuler l'intégration visuelle, et en utilisant un mode de mélange additif. De la valeur obtenue en chaque pixel on peut déduire aisément le nombre de traits se superposant en ce pixel.

3.2.1.2 Les éléments de surface de gravure

Étant donnée la structure des dessins en gravure au burin, l'entité élémentaire de dessin ne pouvait pas être le trait ; il aurait été difficile de gérer autant de primitives. Nous avons choisi de modéliser la gravure par des éléments de surface. Il en résulte que la structure de seuillage a dû être adaptée afin de modéliser de manière réaliste l'interaction outil-matériau.

Comme dans [14] la fonction de seuillage que nous avons utilisé représente une suite de crêtes en forme de triangles (cf. Fig. 3.9) . Les sommets représentant une intensité maximale de gris (du noir) et les creux représentent une intensité minimale (du blanc). Nous utilisons cette fonction analytique pour construire une matrice de seuillage. Cette matrice représente en fait une plaque de métal universelle et lors du seuillage, les différents niveaux d'intensité correspondent à différentes largeurs et profondeurs de traits de gravure au burin.

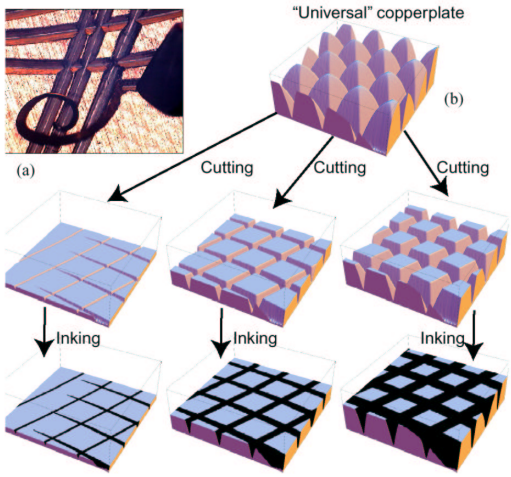


FIG. 3.9 – (a) Microphotographie de la pointe du burin lors du procédé de gravure. (b) La plaque de métal *universelle* coupée à différentes hauteur. Remplir les sillons d'encre imite de manière réaliste le procédé d'impression d'une gravure au burin (illustration tirée de [14].)

Pour pouvoir simuler la gravure, nous avons modélisé l'action de graver comme étant le fait de superposer différentes surfaces paramétriques (comme dans [14]). Nous avons opté pour des surfaces de Bézier, ou *patchs*, qui correspondent à une interpolation bicubique entre 4 courbes de Bézier.

Un élément de surface de gravure est défini par un patch, une référence à la texture de seuillage, une orientation et une fréquence de cette texture, ainsi que par un masque collé sur la surface qui détermine les zones qui seront affectées par ce patch, et enfin par un mode de fusion (minimum ou maximum).

Le patch est contrôlé par seize points ; douze (ceux du bord) reposent sur la surface, et les quatres autres sont déduits par une interpolation linéaire puis décalés afin qu'ils se trouvent *au dessus* de la

surface (comme on peut le voir sur la Fig. 3.10). De cette manière, la texture de seuillage n'est pas directement collée au modèle 3D, ce qui irait à l'encontre des images produites habituellement en gravure : l'orientation des traits contribue à suggérer la géométrie de la surface, mais elle ne doit pas en révéler tous détails sous peine d'être d'une lourdeur peu digeste. L'exhaustivité nuit souvent à la lisibilité et à l'expressivité du dessin.

Un autre argument en faveur d'un détachement des primitives de dessin par rapport à leur support est que le graveur traditionnel travaille dans l'espace image et n'a qu'une idée très vague (du moins personnelle) de la géométrie précise de l'objet qu'il cherche à immortaliser, et par conséquent n'en illustre qu'une approximation grossière. D'autres composantes telles que la gestion des tons (c'est-à-dire l'illumination et la texture), contribuent plus à donner l'impression de volume recherchée.



FIG. 3.10 – Exemple d'un patch de gravure au burin : les douze points de contrôle sur les bords reposent sur la surface, et les quatre du centre sont déduis par une interpolation linéaire puis déplacés au dessus de la surface par une constante controlée par l'utilisateur

À l'opposé des traits, il n'y a pas de gestion de la visibilité entre les patchs de gravure et le modèle. Il ne faut pas perdre de vue que le rôle des primitives de dessin est uniquement de générer une carte de seuillage, qui servira à seuiller la carte des tons cible qui est un rendu du modèle 3D dont la visibilité a été calculée. D'une certaine manière les patchs héritent de la visibilité calculée pour le modèle. Par conséquent ils peuvent très bien se situer en dessous du modèle pourvu que leurs projections dans l'espace image correspondent à ce que désire l'utilisateur.

Il y a un calcul de visibilité intra-patch : un patch peut s'auto-dissimuler. Il est nécessaire de procéder à cette opération car un patch est censé correspondre à une et une seule orientation de gravure dans l'espace image, si l'on ne calcule pas quelles sont les parties cachées, elles se superposent avec les

parties visibles produisant une structure de seuillage comportant plusieurs orientation de gravure, effet très peu satisfaisant visuellement. Par contre, il n'y a pas de calcul de visibilité inter-patch car deux patchs peuvent se superposer.

Pour gérer des superpositions complexes de patchs, le modèle intègre un système de masques : a chaque patch est associé une carte de transparence. Cette carte repose sur le modèle 3D, et est contrôlée de la même manière que les cartes de précision et d'adoucissement (voir les sections 3.2.3.2 et 3.2.3.3).

La carte de recouvrement est calculée de façon analogue que pour les traits.

3.2.2 Opérations de composition

Les mapping représentant les différentes opérations de composition (seuillage, équilibrage et interpolation) ont été implémentés en *hard* sur les stations SiliconGraphics le permettant (Octane2), mais ils ont également été adaptés en *soft* et parallèlisés pour tourner sur des station Unix ne disposant pas d'accéleration graphique matérielle. Le problème de l'accélération matérielle est qu'elle nécessite une fonction de *pixel texture* [9] (rarement présente dans le materiel grand public) qui permet d'utiliser les informations de couleur du *frame buffer* comme coordonnées de texture pour le mapping 3D. Nous obtenons ainsi un mapping très efficace entre un espace de dimension 3 et un espace de dimension 3. C'est pourquoi tous les mappings de notre algorithme sont de dimension inférieure ou égale à trois. Concrètement pour chaque fonction à appliquer (équilibrage ou seuillage), les trois types de donnée en entrée sont stockées dans les trois couches RVB du frame buffer, la texture 3D correspondant à la fonction est chargée puis appliquée à chaque pixel (voir la Fig. 3.11 qui présente la fonction d'équilibrage).

3.2.3 Contrôle de l'utilisateur

Dans cette partie nous étudierons comment l'utilisateur peut dessiner et contrôler le placement des traits, ainsi que l'influence qu'il peut avoir sur la précision et l'adoucissement de l'image finale.

3.2.3.1 Interface de construction des primitives

L'utilisateur dessine directement des traits sur la surface 3D. Nous utilisons une méthode d'optimisation qui force les courbes poly-Bézier à être de continuité G^1. Les points de contrôle de la courbe restent collés sur la surface, même lors d'une édition *a posteriori* (déplacement d'un point de la courbe, ou de la courbe entière). Comme dans tout programme de dessin vectoriel (de type Illustrator), il est possible de rajouter ou d'enlever dynamiquement des points de contrôle. Dans un soucis de clarté de l'interface, les traits peuvent être organisés en couches, afin de faciliter l'édition et la sélection de groupes de traits. De plus, un outil (rudimentaire) de création de hachures est proposé, basé sur une interpolation entre un ou deux traits porteurs et des perturbations aléatoires (voir Fig.3.12).

3.2.3.2 Précision

Le contrôle de la quantité et de la qualité des détails est un aspect primordial lors de la création d'un dessin. Malheureusement (ou heureusement, c'est selon) il n'existe aucune règle précise et formelle qui régisse cet art, que seule l'expérience humaine peut parfois maîtriser. Notre système propose cependant un modèle simpliste de gestion de la précision, qui est controlée spatialement *via* l'édition d'une carte

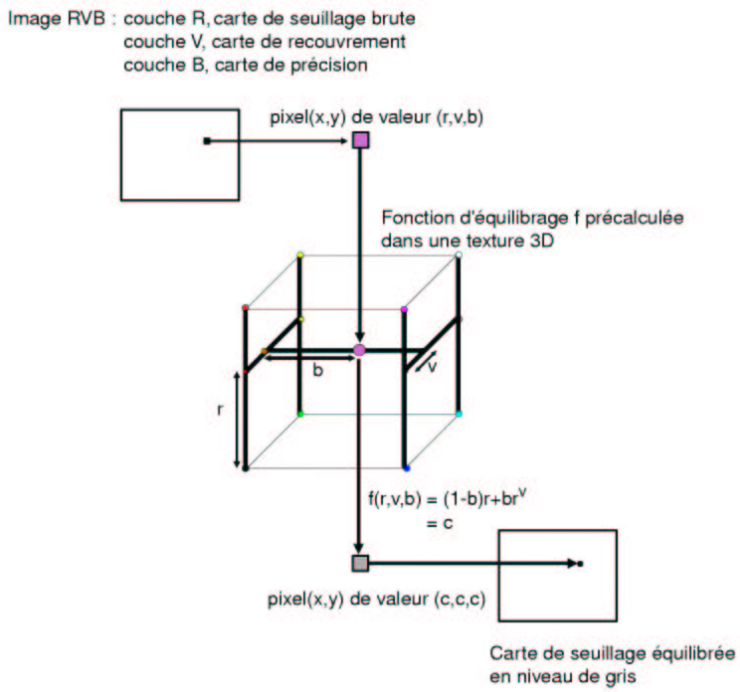


FIG. 3.11 – Expression de la fontion d'équilibrage à l'aide d'une texture 3D qui s'applique à chaque pixel en fonction de ses composantes *RVB*.

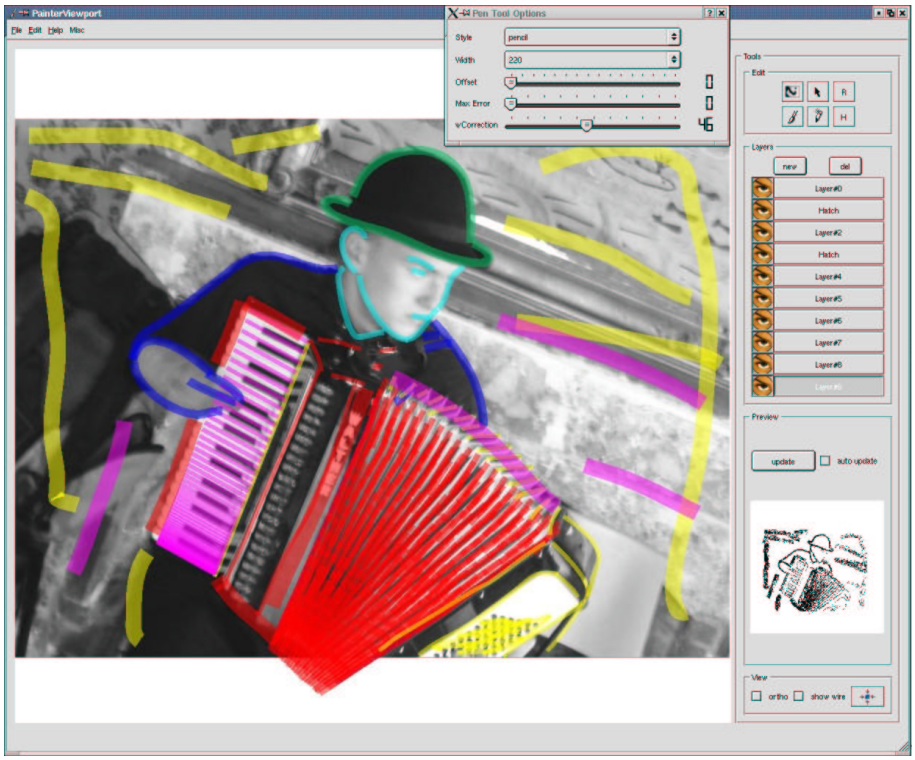


FIG. 3.12 – Session de dessin au trait ; l'ensemble des traits sur l'accordéon a été créé par l'outil de hachure. En bas à droite on peut voir un aperçu du dessin en cours de création.

de précision. Deux mécanismes sont pris en compte et leur influence respective est définie par deux paramètres globaux.

– Le premier mécanisme de gestion de la précision contrôle l'importance de reproduction tonale : il influe sur la quantité d'équilibrage apporté. En effet, des marques de dessin plus saillantes permettent de concentrer l'attention sur les qualités graphiques du dessin, tandis qu'une reproduction fidèle des tons apporte plus de crédibilité dans la reproduction du dessin.
  Une précision de 1 correspond à un équilibrage normal, tandis qu'une précision de 0 correspond à aucun équilibrage appliqué à la texture de seuillage. Ce mécanisme est implémenté en modifiant la fonction d'équilibrage présenté dans la section 3.1.3.2. Comme nous utilisons le mode de mélange *max*, une précision moindre implique que les zones couvertes par plusieurs primitives seront reproduites d'une manière plus sombre. Cela signifie que le ton de sortie est alors plus déterminé par le nombre de primitives se recouvrant que par la carte des tons cibles.
– Le deuxième mécanisme contrôle la quantité spatiale de détails présente dans la carte des tons cibles. Dans le processus de création d'un dessin, il est courant d'utiliser des tons uniformes dans les régions sans importances, et de concentrer les détails sur les zones où l'on souhaite focaliser l'attention. Cette fonctionalité aurait pu être réalisée en utilisant un filtre de floutage non linéaire dont la taille du noyau gaussien aurait été une fonction de la carte de précision ; mais cette solution serait nettement trop coûteuse en calculs pour une implémentation en temps réel. Nous effectuons donc une approximation en utilisant une interpolation linéaire entre la version originale de l'image de référence et une version basse résolusion filtrée avec une gaussienne (dans le même esprit que [11]).

Les cartes de précision et d'adoucissement disposent de la même interface d'édition : l'utilisateur dessine directement à l'aérographe sur la surface 3D, et les valeurs correspondantes dans l'espace de la texture sont mise à jour. Il est important de souligner que ces cartes sont définies dans l'espace objet, et directement attachées à la surface, ce qui permet éventuellement de détacher les primitives de dessin de cette surface (c'est le cas pour la gravure au burin, cf. section 3.2.1.2). En effet il est logique que les notions de précision et d'adoucissement soient liées aux propriétés géométriques du modèle, et non aux primitives qui ont pour but de les illustrer dans l'espace image.

3.2.3.3 Adoucissement

Une technique traditionelle du dessin consiste à adoucir l'image en la frottant (avec les doigts ou d'autres outils), afin d'étaler les pigments de l'encre ou les particules de graphite.

Contrairement à la technique utilisée par Sousa et Buchanan [20], l'implémentation de l'adoucissement que nous utilisons est purement phénoménologique. C'est une extension de la technique d'antialiasing (décrite dans la partie 3.1.3.1) qui consiste à faire varier la largeur de la fonction s', utilisée pour le seuillage.

Chapitre 4

Résultats

4.1 Dessin au trait

Lorsque l'on ne dispose d'aucune information tridimensionnelle sur l'image à traiter on obtient le même type de résultats que [2] : un dessin en deux dimensions. Les figures 4.1 et 4.2 illustrent les différents types rendu que l'on peut obtenir en quelques minutes d'utilisation du système.



FIG. 4.1 – Dessin issu de la session présentée dans la Fig. 3.12. Rendu au crayon

On peut voir sur ces dessins l'effet de la texture de seuillage sur le rendu. Dans le cas du dessin au fusain (Fig. 4.1), la largeur de chaque trait reste constante, les tons plus clairs sont exprimés en diminuant la densité, comme si on exerçait une pression moindre sur l'outil.

Dans le cas de la pointe sèche, les tons plus clairs sont exprimés en diminuant la largeur du trait, ce qui permet quelques effets subtils comme on peut le voir avec les touches de l'accordéon dans la Fig.



FIG. 4.2 – En changeant la texture de seuillage pour simuler un aspect de dessin à la pointe sèche

4.2, ou bien les traits dans le cou de la Fig. 4.3.

Cependant, l'objectif du système est de pouvoir expérimenter ce que pourraient être des dessins en 3D. Nous ne prétendons pas répondre à cette question qui est largement ouverte et qui regroupe un certain nombre de problèmes non triviaux tels que le comportement de la forme et du mouvement des traits par rapport à un changement de point de vue, ou bien les changements d'échelles de texture lors d'un zoom. Nous souhaitons juste dans un premier temps apercevoir quel est le champs des possibilités.

La Fig. 4.4 présente les résultats que l'on peut obtenir lorsque l'on dispose de données 3D associées à la photo (ici une carte de profondeur, faite séparément et manuellement). Les traits ont été placés par l'utilisateur sur le visage, et on peut se déplacer interactivement autour du «dessin»(voir en bas de la figure). Il est facilement possible de déplacer les traits pour qu'ils correspondent mieux à un certain point de vue, mais cela n'a pas été fait dans cet exemple, d'où des dessins peu crédible que l'on a dans certains cas (en bas à droite de la figure, lorsque l'on s'éloigne du point de vue de référence).

Après observation, il apparait qu'un système de dessin en 3D devrait pouvoir gérer deux types de traits :

- des traits dynamiques, dont la position et la forme dans l'espace objet seraient une fonction du point de vue, tels que les traits de silhouette.
- des traits fixes, dont le role est de faire ressortir des particularités géométriques du modèle et donc attachés d'une manière plus ou moins solide à celui-ci.

4.2 Gravure au burin

La Fig. 4.5 montre l'effet que l'on obtient en utilisant un seul patch pour la gravure au burin. Le patch englobe tous les détails importants (nez, yeux, bouche) de l'objet géométrique, ce qui permet de



FIG. 4.3 – Dessin à la pointe sèche. La largeur des traits est proportionelle à l'intensité des tons.

garder une carte de seuillage cohérente lorsque le point de vue change.

Lorsque l'on tourne autour de l'objet, il apparait naturel que la texture de seuillage ne soit pas attachée directement au modèle : cela laisse l'impression que le dessin est bien une image créée en 2D.

Il reste des problèmes d'équilibrage lors de la superposition de patchs comme l'on peut le voir sur les figures 4.7 et 4.6. Les zones où les deux patchs se recouvrent sont globalements assombries. Pour des raisons obscures, le modèle d'équilibrage probabiliste présenté dans la partie 3.1.3.2 ne marche pas aussi bien que celui utilisé pour les traits.

Les résultats que nous obtenons pour la gravure sont d'une qualité moins remarquable que ceux présentés par Ostromoukhov dans [14]. Ce n'est pas dû au modèle : l'implémentation réalisée dans une optique d'interactivité effectue de trop grosses approximations, dont certaines (l'équilibrage principalement) peuvent être améliorées.

L'orientation des traits de gravure joue un role important dans la perception des formes. Comme le montre la Fig. 4.8, il semblerait qu'une orientation oblique des traits soit mieux à même de retransmettre les différentes caractéristiques du dessin.

Bien que ce soit d'un intérêt limité pour l'instant, il est possible de mélanger les deux types de primitives de dessin, c'est ce que présente la Fig. 4.9.

4.3 Limitations

La principale limitation de notre modèle vient du support du dessin : l'image de référence. Les tons étant obtenus par cette dernière, les résultats sont fortement dépendants de sa qualité. Une photographie de piètre qualité, tant dans la composition que dans l'éclairage (contraste et luminosité) ne pourra jamais être un bon support pour le dessin. La méthode de seuillage que nous utilisons nécessite des images bien



FIG. 4.4 – Dessin au trait en 3 dimensions.

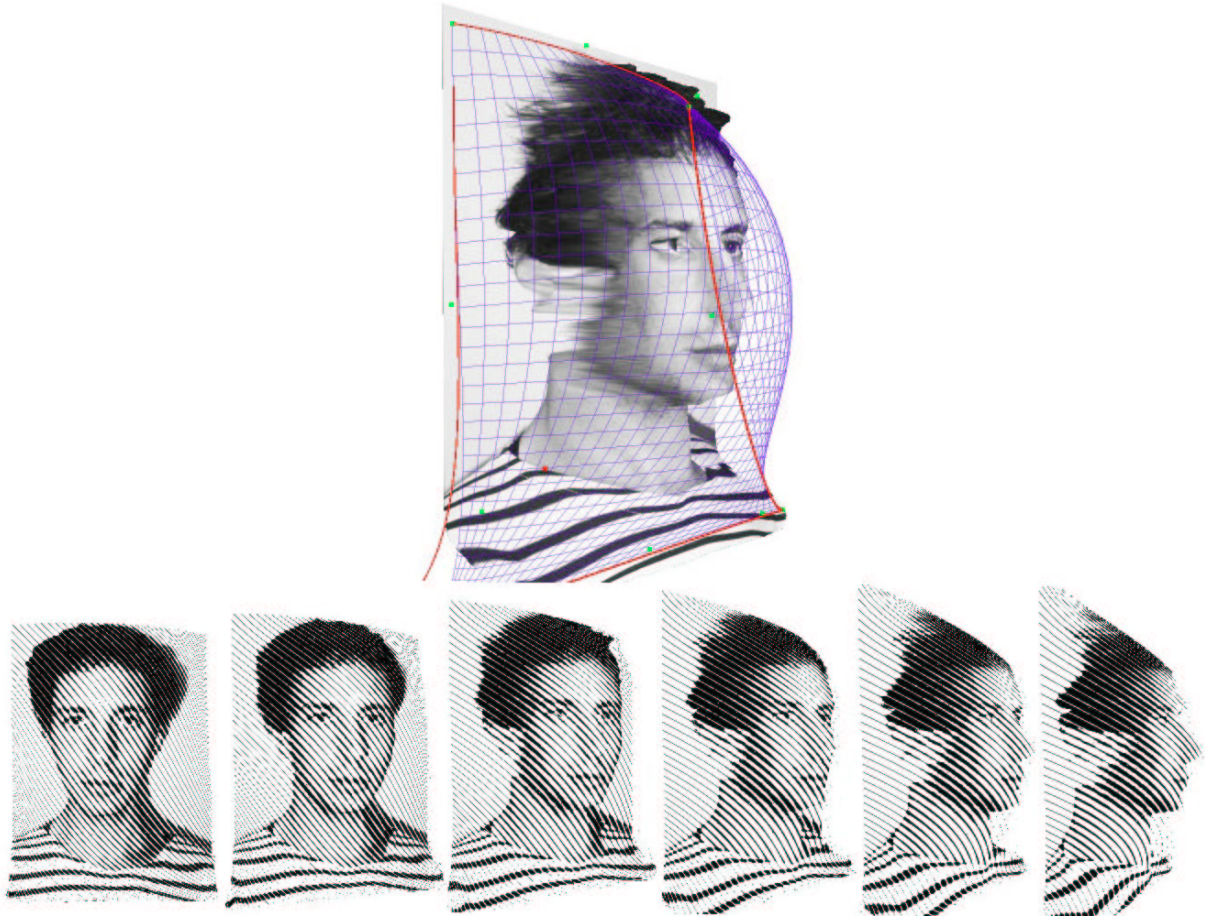


FIG. 4.5 – Gravure en 3 dimensions.



FIG. 4.6 – Gravure au burin d'un visage en utilisant deux patchs.

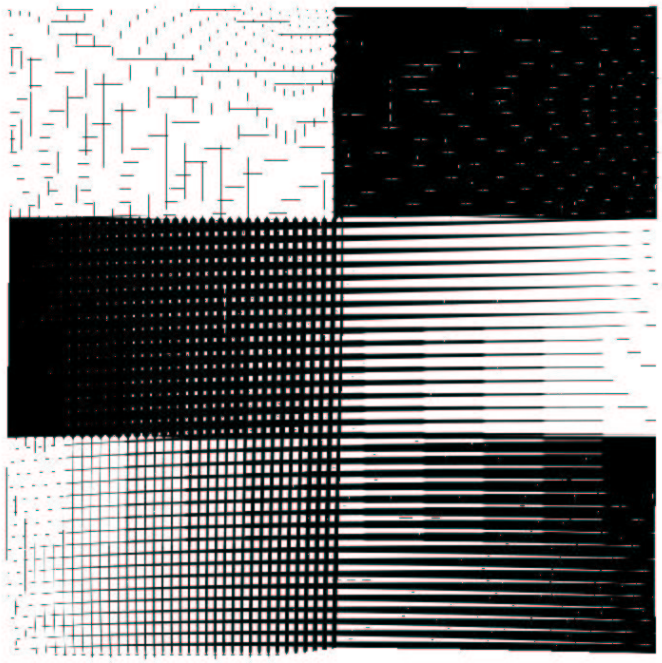


FIG. 4.7 – Problème d'équilibrage lors de la superposition de deux patchs. Sur la partie gauche de l'image, les tons sont globalement assombris, par rapport à la droite où il n'y a qu'un seul patch.



FIG. 4.8 – Rendu d'un même objet avec différentes orientations de gravure.



FIG. 4.9 – Utilisation de différents types de primitives au sein d'un même dessin.

contrastées, afin de reproduire précisément les détails.

L'implémentation du modèle que nous proposons pour le dessin au trait en 3D est également limité par la nature statique des traits qui le compose. Mais rien n'empêche l'intégration de règles de déplacement de traits, et de détection automatique de contours dans le modèle, si ce n'est le manque de temps.

4.4 Améliorations

À court terme, de nombreuses améliorations sont envisageables, afin de palier les défauts présents dans les résultats. Pour générer des traits dynamiquement, deux approches sont envisageables :

- ajouter un système de détection de silhouette : de nombreuses recherches ont été faite dans ce domaine récemment ; il serait donc possible de générer automatiquement des traits correspondants à la silhouette en temps réel.
- un système d'interpolation des traits qui ont été définis en différents points de vue pourrait compléter les traits de silhouette. Mais le problème de l'interpolation de traits reposants sur un objet quelconque est loin d'être trivial. Il faudrait alors passer par un autre objet approximant de façon grossière le modèle 3D et correctement paramétrisé. Les traits seraient interpolés sur cette surface puis projeté pour le rendu sur l'objet initial.

Une autre amélioration possible serait d'utiliser les patchs comme support pour d'autres textures de seuillage, qui pourraient simuler d'autres techniques traditionelles.

Chapitre 5

Conclusion

Nous avons présenté un système de dessin interactif tridimensionnel basé sur un modèle de seuillage des entités de dessin. Cette technique permet de simuler de manière convaincante différentes techniques traditionelles de dessin avec une grande souplesse et de manière interactive. Elle permet également de réaliser la séparation des aspects esthétiques et techniques lors du processus création. L'utilisateur garde cependant un contrôle permanent, tant à haut niveau, par exemple en précisant localement la quantité de détail, qu'à bas niveau.

Dans le cours délais qu'autorise le temps d'un stage de DEA, j'ai essentiellement travaillé sur une réimplémentation complète du modèle de Durand et Ostromoukhov, ainsi qu'à son extension. Cependant quantités de développement sont possibles à moyen ou long terme.

On peut citer notament :

- Étudier les effets du positionnement et de la forme des traits sur notre perception d'un dessin représentant une scène 3D.
- Mieux comprendre les règles qui font que l'on trouve certaines images plus plaisantes que d'autre ; et les appliquer à l'infographie.
- Tenter de formaliser quels sont les différents styles de dessin, en fonction d'un certain nombre de paramètres.
- la gestion de la couleur est un sujet interressant mais très délicat, car nous pensons qu'il serait réducteur de la traiter simplement comme les tons.

Remerciements À Joëlle Thollot, Victor Ostromoukhov et Frédo Durand pour leur aide et leur soutien précieux. Et à Nicolas Holzschuch pour sa patience...

Bibliographie

[1] CURTIS (Cassidy), ANDERSON (Sean), FLEISCHER (Josh Seims Kurt), et SALESIN (David), « Computer-Generated Watercolor », *in Computer Graphics proceedings, SIGGRAPH'97*, Annual Conference series, 1997, http ://www.cs.washington.edu/homes/cassidy/watercolor/.

[2] DURAND (Frédo), OSTROMOUKHOV (Victor), MILLER (Mathieu), DURANLEAU (François), et DORSEY (Julie), « Decoupling Strokes and High-Level Attributes for Interactive Traditional Drawing », *in Eurographics Workshop on Rendering 2001*, 2001.

[3] EDWARDS (Betty), *The New Drawing on the Right Side of the Brain*, J P Tarcher, 1999.

[4] ELBER (Gershon), « Line art rendering via a coverage of isoparametric curves », *IEEE Transactions on Visualization and Computer Graphics*, 1, n° 3, septembre 1995, p. 231–239, ISSN 1077-2626.

[5] _____, « Interactive Line Art Rendering of Freeform Surfaces », *Computer Graphics Forum*, 18, n° 3, septembre 1999, p. 1–12, ISSN 1067-7055.

[6] GAIR (Angela), *Artist's Manual : A Complete Guide to Painting and Drawing Materials and Techniques*, Chronicle Books, 1996.

[7] GOOCH (Bruce), COOMBE (Greg), et SHIRLEY (Peter), « Artistic Vision : Painterly Rendering Using Computer Vision Techniques », *in ACM/Eurographics Symp. NPAR'02*, June 2002.

[8] HAEBERLI (Paul), « Paint By Numbers : Abstract Image Representations », volume 24, p. 207–214, août 1990.

[9] HEIDRICH (W.), WESTERMANN (R.), SEIDEL (H.-P.), et ERTL (T.), « Applications of Pixel Textures in Visualization and Realistic Image Synthesis », *in ACM Symposium on Interactive 3D Graphics*, ACM/Siggraph, 1999.

[10] HERTZMANN (Aaron) et ZORIN (Denis), « Illustrating Smooth Surfaces », *in* AKELEY (Kurt), éditeur, *Siggraph 2000, Computer Graphics Proceedings*, p. 517–526, ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.

[11] KAUTZ (K.), VAZQUEZ (P.), HEIDRICH (W.), et SEIDEL (H.), « A Unified Approach to Prefiltered Environment Maps », 2000.

[12] LEISTER (W.), « Computer generated copper plates », *Computer Graphics Forum*, 13, n° 1, mars 1994, p. 69–77.

[13] MARKOSIAN (Lee), KOWALSKI (Michael A.), TRYCHIN (Samuel J.), BOURDEV (Lubomir D.), GOLDSTEIN (Daniel), et HUGHES (John F.), « Real-Time Nonphotorealistic Rendering », *in* WHITTED (Turner), éditeur, *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, p. 415–420, ACM SIGGRAPH, Addison Wesley, août 1997, ISBN 0-89791-896-7.

[14] OSTROMOUKHOV (Victor), « Digital Facial Engraving », *in* ROCKWOOD (Alyn), éditeur, *Siggraph 1999, Computer graphics Proceedings*, Annual Conference Series, p. 417–424, ACM Siggraph, Los Angeles, 1999.

[15] OSTROMOUKHOV (Victor) et HERSCH (Roger D.), « Multi-Color and Artistic Dithering », *in* ROCKWOOD (Alyn), éditeur, *Siggraph 1999, Computer Graphics Proceedings*, Annual Conference Series, p. 425–432, Los Angeles, 1999, ACM Siggraph, Addison Wesley Longman.

[16] SAITO (Takafumi) et TAKAHASHI (Tokiichiro), « Comprehensible Rendering of 3-D Shapes », volume 24, p. 197–206, août 1990.

[17] SALISBURY (Michael P.), ANDERSON (Sean E.), BARZEL (Ronen), et SALESIN (David H.), « Interactive Pen–And–Ink Illustration », *in* GLASSNER (Andrew), éditeur, *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, p. 101–108, ACM SIGGRAPH, ACM Press, juillet 1994, ISBN 0-89791-667-0.

[18] SALISBURY (Michael P.), WONG (Michael T.), HUGHES (John F.), et SALESIN (David H.), « Orientable Textures for Image-Based Pen-and-Ink Illustration », *in* WHITTED (Turner), éditeur, *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, p. 401–406, ACM SIGGRAPH, Addison Wesley, août 1997, ISBN 0-89791-896-7.

[19] SOUSA (Mario Costa) et BUCHANAN (John W.), « Observational Models of Graphite Pencil Materials », *in* DUKE (David), COQUILLART (Sabine), et HOWARD (Toby), éditeurs, *Computer Graphics Forum*, volume 19(1), p. 27–49, Eurographics Association, 2000.

[20] SOUSA (Mario Costa) et BUCHANAN (John William), « Observational Model of Blenders and Erasers in Computer-Generated Pencil Rendering », *in Graphics Interface*, p. 157–166, juin 1999.

[21] TAKAGI (Saeko), FUJISHIRO (Issei), et NAKAJIMA (Masayuki), « Volumetric modeling of artistic techniques in colored pencil drawing », *in* ACM, éditeur, *SIGGRAPH 99. Proceedings of the 1999 SIGGRAPH annual conference : Conference abstracts and applications*, Computer Graphics, p. 283–283, New York, NY 10036, USA, 1999, ACM Press.

[22] ULICHNEY (Robert), *Digital Halftoning*, Cambridge, MA, USA, MIT Press, 1987.

[23] WEBB (Matthew), PRAUN (Emil), FINKELSTEIN (Adam), et HOPPE (Hughes), « Fine Tone Control in Hardware Hatching », *in Proceedings of NPAR 2002, International Symposium on Non Photorealistic Animation and Rendering (Annecy, France, juin2002)*, 2002.

[24] WINKENBACH (Georges) et SALESIN (David H.), « Computer–Generated Pen–And–Ink Illustration », *in* GLASSNER (Andrew), éditeur, *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, p. 91–100, ACM SIGGRAPH, ACM Press, juillet 1994, ISBN 0-89791-667-0.