



## Traitement, Synthèse, Technologie et Applications

BIARRITZ - Mai 1984 -

SYNTHESE D'IMAGES AVEC TEXTURES STRUCTUREES ET EFFETS PARTICULIERS

IMAGE GENERATION WITH STRUCTURED TEXTURES AND SPECIAL EFFECTS

A. BRUNO , D. BARBA

LABORATOIRE D'AUTOMATIQUE , I.N.S.A.  
20, AVENUE DES BUTTES DE COESMES - 35043 RENNES CEDEX**RESUME**

Les images synthétiques comportent encore des défauts de qualité et de réalisme par comparaison aux images naturelles lorsqu'elles sont engendrées avec des contraintes de temps réel. Nous présentons des méthodes compatibles avec ces contraintes et qui permettent d'améliorer le réalisme des images engendrées par ordinateur.

Une première amélioration est apportée par l'ajout de texture sur les facettes des objets à représenter. Les variations d'ombrage à l'intérieur des facettes peuvent être plus ou moins structurées et complexes pour améliorer le réalisme. Ces variations d'ombrage qui constituent des motifs, représentent ce que nous appelons la texture structurée. La représentation économique de celle-ci dans les bases de données est compatible avec les contraintes de temps réel et de cohérence de texture d'image à image.

L'affichage d'images texturées présente des défauts de type moirage si le signal de texture est sous-échantillonné. Une méthode simple et efficace basée sur un filtrage bidimensionnel adaptatif évite ces défauts de moirage.

Les effets particuliers que nous proposons apportent une amélioration évidente du réalisme des images synthétiques dans une large gamme d'applications (simulation, architecture, dessin animé, ...). L'effet de transparence permet la visualisation d'objets situés derrière une facette transparente en tenant compte des caractéristiques optiques de cette dernière. Plusieurs facettes peuvent être superposées simultanément. Pour l'effet de brume, nous prenons en compte la distance objet-observateur ainsi que la densité de la brume. La production de ces deux effets est compatible avec l'implantation de textures.

**SUMMARY**

Real time synthetic images contain defects of quality and realism compared with natural images. We present real time methods which enhance the realism of computer-generated images.

A first improvement is got with the mapping of textures on to the objects' facets to display. The shading inside each facet can support structured and complex variation in order to realize a good realism. These shading variations give patterns representing what we call structured textures. A space-saving representation of these textures in data bases is consistent with real time algorithms and texture coherence in successive frames.

Textured images are displayed with moire patterns if the texture signal is under-sampled. A simple and efficient method using adaptive bidimensional filtering techniques remove these artifacts.

The special effects we propose, enhance realism of synthetic images within many applications (simulation, architecture, cartoons, ...). The transparency effect is used for displaying objects behind a transparent facet and taking up of the optical properties of that facet. This effect remains with several facets displayed simultaneously. The haze effect is realized taking up the object-observer distance and the haze 'opacity. These two effects are consistent with the mapping of textures and real time constraints.



## I - INTRODUCTION

La synthèse d'images par ordinateur, née il y a un peu plus d'une décennie, connaît aujourd'hui des applications dans des domaines très variés : les centres de simulation [1] (images aériennes, images spatiales, ...), l'architecture, le génie civil, la conception assistée par ordinateur, le génie médical, la cartographie [2], le dessin animé, la fabrication de films publicitaires, ... Les images synthétiques doivent reconstituer pour un observateur l'environnement visuel correspondant à une application donnée. Des images de très bonne qualité et d'un bon réalisme ont été synthétisées mais leur coût est très élevé car les temps de calcul sont de plusieurs minutes, voire fréquemment de plusieurs heures. Cela interdit toute interactivité avec l'utilisateur (pilote dans un simulateur d'avion, ingénieur en C.A.O. par exemple). Il est intéressant de rechercher un compromis qui permet la synthèse en temps réel ou quasi temps réel de scènes de complexité moyenne tout en conservant une qualité et un réalisme tels que toute l'information pertinente contenue dans la scène soit perçue par l'utilisateur sans aucune gêne visuelle marquée.

Une amélioration importante du réalisme des images synthétiques est obtenue par l'ajout de texture sur les facettes des objets. Les ombrages de type lissé (à variation linéaire) concernent surtout des objets artificiels, tous les autres objets ont des ombrages, soit qui font apparaître des motifs plus ou moins bien fixés et ordonnés spatialement (textures structurées), soit pour lesquels l'évolution spatiale paraît plutôt désordonnée (textures aléatoires). Un certain nombre de modèles utilisés pour la synthèse de textures aléatoires ont été présentés ou mis en oeuvre dans [3]. Pour les textures structurées, nous utilisons un modèle par graphe qui permet leur représentation dans les bases de données de façon économique. Les textures obtenues par synthèse sont mémorisées dans des tables dites tables de texture. Chaque table contient une texture normalisée (sans déformation géométrique et avec un contraste correspondant à un éclairage perpendiculaire au plan de la texture).

En synthèse d'image, un des principaux défauts de qualité à combattre est le phénomène d'"aliasage" qui se manifeste déjà dans les images fixes et dont la perception est encore accentuée dans les séquences d'images animées. Les contours des objets apparaissent comme des escaliers qui sont animés de mouvements plus ou moins réguliers lorsque les objets sont en mouvement. Ceci est dû à l'interaction entre la position des contours, la structure spatio-temporelle de l'image et les caractéristiques spatio-temporelles du filtrage introduit par le système visuel humain.

Les méthodes de traitement "anti-aliasage" que nous avons développé dans [4] sont basées sur un filtrage spatial monodimensionnel adaptatif. Des défauts de même nature apparaissent dans les zones d'images texturées. Ces défauts appelés moirage, sont parfois aussi gênants que les précédents car le contenu sémantique apporté par la texture d'un objet ou d'une facette peut être totalement perdu (dans le cas d'un contour, l'information apportée par celui-ci n'est jamais totalement perdue). Un filtrage bidimensionnel adaptatif limité au seul signal de texture permet d'éliminer ces défauts de moirage.

Parmi les nombreux effets spéciaux possibles (transparence, ombres portées, réflexions, brume, ...) l'effet de transparence peut apporter un réalisme accru dans un certain nombre d'applications (architecture, C.A.O. par exemple) sans être très coûteux en temps calcul. Cet effet permet la visualisation d'objets masqués en totalité ou en partie par d'autres objets. Le calcul de l'intensité lumineuse est effectué en tenant compte des caractéristiques optiques des facettes transparentes. Les facettes transparentes

situées entre la source de lumière et la facette en cours de génération ne sont pas prises en compte pour le calcul de l'intensité lumineuse de cette facette, cet effet est à relier directement à celui des ombres portées qui n'est pas traité ici. L'effet de brume est un effet de même nature que l'effet de transparence. Le réalisme apporté par cet effet pour les simulateurs (avions, bateaux) est excellent. Les caractéristiques de luminance, teinte et densité de la brume sont prises en considération avec la distance objet-observateur pour le calcul de l'atténuation.

## II - GENERATION D'IMAGES SYNTHETIQUES AVEC TEXTURES

### 1) Modèle d'images d'objets

Le modèle d'objet que nous avons utilisé est à facettes planes polygonales. Cette représentation à facettes planes permet la synthèse en temps réel avec des algorithmes simples lorsque les surfaces des objets ne sont pas texturées. Cette même représentation est bien adaptée à la modélisation du signal image sous la forme de deux composantes additives :  $I = I_p + I_T$  où la première composante  $I_p$  est appelée composante par plateau et la seconde composante  $I_T$  est la composante de texture. Nous avons supposé que la composante  $I_p$  porte des informations de luminance, teinte et saturation. Les variations de luminance à l'intérieur d'une même facette, sont supportées par les composantes  $I_p$  et  $I_T$ .

La composante par plateau  $I_p$  supporte l'information de luminance moyenne locale du signal  $I$  et de sauts de valeur moyenne (contours) dans l'image. La composante de texture  $I_T$  correspond aux variations du signal  $I$  autour de sa valeur moyenne locale  $I_p$ . Elle est associée aux caractéristiques visuelles de la surface de chacune des facettes représentant les objets, elle apporte un contenu sémantique important : orientation de la facette déduite des gradients présents dans la texture, nature physique du matériau constituant l'objet par exemple.

Les algorithmes de synthèse d'images utilisant le modèle à facettes planes permettent la synthèse de la composante par plateau de façon très simple. La composante de texture nécessite l'utilisation de méthodes de synthèse plus complexes, ses variations étant généralement très complexes à l'intérieur de la facette à engendrer. Ces méthodes qui sont coûteuses en temps calcul peuvent être appliquées "off-line" pour créer un échantillon de texture qui est conservé dans une table. Il suffit alors de lire cette table de texture pour obtenir le signal texture en temps réel. Pour cela, il faut d'une part calculer l'adresse de chaque point image de la facette dans la table de texture (pour prendre en compte l'effet de perspective le calcul est relativement coûteux) et, d'autre part, réaliser un filtrage bidimensionnel de la texture (plusieurs points de la texture peuvent être représentés par le même point image) pour obtenir une qualité visuelle suffisante, notamment lors des séquences d'images animées. Les méthodes de génération de la composante texture contenue dans la table de texture font l'objet du paragraphe suivant. Le traitement anti-moirage appliqué sur le signal texture est ensuite brièvement décrit.

### 2) Modèles de textures structurées

Dans le cas des images numériques, la texture peut être caractérisée par la distribution spatiale du signal d'image dans un ensemble de points appartenant à une certaine zone connexe dans l'image. La texture est dite structurée lorsque cette distribution présente certaines régularités spatiales observables, dans le cas contraire, la texture est dite aléatoire. Une texture structurée peut être décrite par un ensemble de primitives et par leurs règles de placement, les primitives

SYNTHESE D'IMAGES AVEC TEXTURES STRUCTUREES ET EFFETS PARTICULIERS  
 IMAGE. GENERATION WITH STRUCTURED TEXTURES AND SPECIAL EFFECTS

A. BRUNO , D. BARBA

d'une même classe présentant toutes des propriétés de régularité identique (valeur moyenne identique, forme de primitive identique par exemple).

Ehrich et Foith [5] ont développé un modèle simple utilisant un graphe unique pour assembler les primitives. Les primitives utilisées correspondent aux pics de luminance et leur voisinage (maxima locaux entourés de tous côtés par des minima locaux). Ce modèle n'est pas adapté pour la représentation des primitives en vue de leur synthèse mais plutôt en vue de leur analyse, notamment pour certains types de textures (forêt dont les arbres sont discernables individuellement par exemple).

La nécessité de décrire chaque primitive avec un graphe devient inévitable si l'on veut produire des primitives plus complexes que les précédentes. Yokoyama [6] a été l'un des premiers à utiliser un graphe pour la description de chaque primitive. Une image symbolique est engendrée en distribuant des noeuds à partir desquels on développe les graphes correspondant aux primitives. L'image numérique est obtenue par transformation de chaque symbole en une valeur de luminance. Des modifications de cette image donnent un ensemble d'images dont les primitives de la texture sont semblables à celles de l'image initiale. Des déformations peuvent être introduites si on utilise des transformations aléatoires plutôt que déterministes. Ce modèle est adapté à la représentation de certaines textures structurées déterministes ou aléatoires. Cependant l'assemblage des primitives n'est pas aussi bien contrôlé que la synthèse des primitives car aucun graphe n'est utilisé pour cet assemblage.

Vilnrotter [7] a utilisé un graphe déterministe pour assembler les primitives. Les primitives utilisées sont simplement des zones connexes à géométrie simple et à luminance uniforme. Ce modèle assez simple pour les primitives est surtout intéressant pour la recherche de l'existence des primitives et l'extraction des règles de placement des primitives dans une texture naturelle.

L'utilisation d'un modèle avec deux niveaux de graphes permet de décrire à la fois des règles de placement plus complexes que celles utilisées par Yokoyama et des primitives plus variées que celles utilisées par Vilnrotter. Zucker [8], puis Lu et Fu [9], [10] ont utilisé ce modèle avec des approches quelque peu différentes au niveau du graphe décrivant l'assemblage des primitives. Dans le modèle de Zucker, chaque primitive déformée est engendrée par un graphe particulier. Le graphe de la texture observable est obtenu par déformation du graphe de la texture idéale, celle-ci étant construite par un assemblage régulier des primitives. Le modèle de Lu et Fu est plus riche car il permet des déformations mieux contrôlées d'une part, et une description plus compacte des règles de placement d'autre part. Les déformations de la texture sont obtenues par modification du graphe qui assemble les primitives. Une grammaire stochastique unique permet la représentation d'un ensemble de graphes d'une façon simple et très condensée. Les primitives sont aussi représentées par une grammaire stochastique. Il est alors possible d'engendrer des textures structurées déformées et bruitées avec un bon contrôle des déformations et du bruit introduit. Des textures structurées binaires (avec deux niveaux de luminance seulement pour toute la texture) ont ainsi été produites avec des primitives plus ou moins déformées.

3) Représentation et synthèse des textures structurées par graphe semi-aléatoire

Le modèle que nous présentons est un modèle avec deux niveaux de graphe qui se différencie des modèles précédents par le fait que, d'une part les graphes possèdent la même structure dans les deux niveaux, et

d'autre part, les possibilités de déformations et de bruits introduits sont plus riches. Une plus grande diversité de primitives peut alors être obtenue à partir d'une seule grammaire.

Chaque niveau du graphe correspond à un élément de base de la texture : au niveau le plus élevé, se trouve le médaillon, et au niveau inférieur la primitive. Celle-ci est obtenue avec le "graphe primitive" dont chaque noeud est associé, dans la texture normalisée, à un point d'image de la texture pour lequel nous définissons les variations de luminance et de position par rapport au point image associé au noeud père dans le graphe. Le graphe est déterministe ou semi-aléatoire suivant que nous associons à chaque noeud un ensemble unique ou plusieurs ensembles de valeurs pour les variations de luminance et de position. Dans ce dernier cas, à chaque ensemble de valeurs luminance-position est associée une probabilité. A partir d'une description unique et compacte, nous pouvons obtenir des motifs très variés dont les formes et les distances de corrélation pour l'ombrage sont facilement contrôlables. Le graphe d'une primitive déterministe est donné en figure 1 avec l'image de la primitive obtenue. La signification des notations utilisées est la suivante :

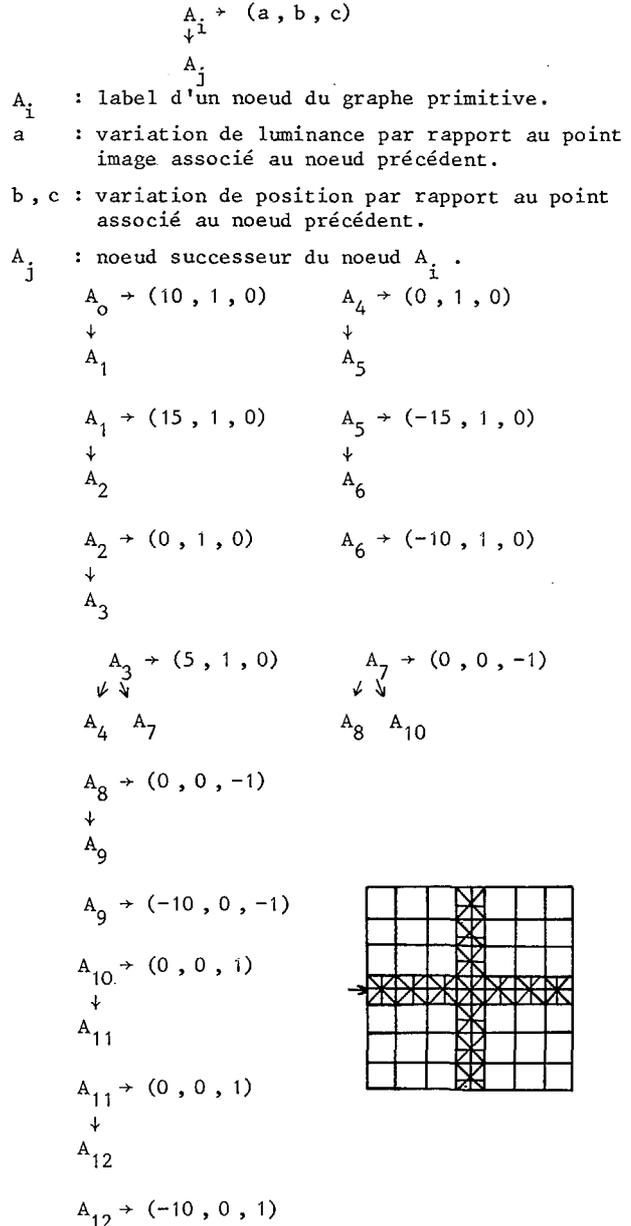


Figure 1 : primitive déterministe : graphe et primitive



SYNTHESE D'IMAGES AVEC TEXTURES STRUCTUREES ET EFFETS PARTICULIERS  
 IMAGE GENERATION WITH STRUCTURED TEXTURES AND SPECIAL EFFECTS  
 A. BRUNO , C. BARBA

La figure 2 représente le graphe semi-aléatoire pour une primitive aléatoire avec quelques réalisations de cette primitive. Les notations utilisées ont la signification précédente avec en plus une probabilité associée à chaque triplet (a, b, c).

- $A_0 \rightarrow (10,1,0),0.4 ; (10,1,1),0.3 ; (10,1,-1),0.3$
- ↓
- $A_1$
- ↓
- $A_2 \rightarrow (15,1,0),1.$                        $A_4 \rightarrow (0,1,0),1.$
- ↓
- $A_3$      $A_5$
- ↓
- $A_2 \rightarrow (0,1,0),1.$                        $A_5 \rightarrow (-15,1,0),1.$
- ↓
- $A_3$      $A_6$
- ↓ ↓
- $A_3 \rightarrow (5,1,0),1.$                        $A_6 \rightarrow (-10,1,0),1.$
- ↓ ↓
- $A_4$   $A_7$
- ↓ ↓
- $A_7 \rightarrow (-5,0,-1),0.4 ; (-10,-1,-1),0.3 ; (-15,1,1),0.3$
- ↓ ↓
- $A_8$   $A_{10}$
- ↓
- $A_8 \rightarrow (-5,0,-1),1.$
- ↓
- $A_9$
- ↓
- $A_9 \rightarrow (-5,0,-1),0.5 ; (-10,0,-1),0.5.$
- ↓
- $A_{10} \rightarrow (0,0,1),1.$
- ↓
- $A_{11}$
- ↓
- $A_{11} \rightarrow (-5,0,+1),1.$
- ↓
- $A_{12}$
- ↓
- $A_{12} \rightarrow (-5,0,+1),0.5 ; (-10,0,1),0.5.$

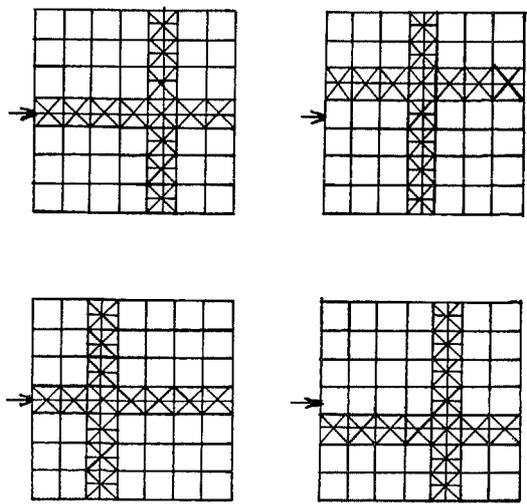


Figure 2 : primitive aléatoire : graphe et différentes primitives obtenues

Le "graphe médaillon" assemble les primitives pour former le médaillon, échantillon de la texture qui va remplir la table de texture normalisée. Comme le graphe primitive, ce graphe peut être de type semi-aléatoire pour introduire des déformations sur la position et la luminosité des primitives les unes par rapport aux autres. Lorsque la texture est constituée d'une seule primitive, un assemblage en parallélogramme est utilisé. La description de l'assemblage est rapide puisqu'il suffit de donner les distances entre primitives suivant deux directions distinctes et l'angle entre ces directions. Si le graphe est semi-aléatoire, nous donnons les valeurs moyennes et écart-types des distances et angle (nous avons utilisé seulement une loi de probabilité gaussienne). Un problème important à résoudre dans ce dernier cas est introduit par la dimension réduite de la table de texture utilisée. La table est de dimension 64x64 et les primitives peuvent atteindre des tailles 8x8, 16x16, voire 32x32, le nombre de primitives dans la table de texture devient donc réduit et la périodicité due à la table de texture devient perceptible même si lors de la visualisation de la facette, l'effet de perspective réduit la perception visuelle de telles périodicités. Une première possibilité consiste à augmenter la taille de la table de texture à 128x128 ou 256x256 mais le coût mémoire d'une telle opération est à rejeter s'il est nécessaire d'utiliser un nombre important de textures différentes. Une autre possibilité pour supprimer l'effet de périodicité de la table de texture, ou plus exactement les défauts apportés par les bords de la table de texture, est de prendre en compte les relations de voisinage qui existe entre les primitives dans la texture lorsque celle-ci est implantée sur une facette et non pas dans la table de texture. Il faut alors considérer que chaque primitive possède quatre primitives voisines dans la texture, quel que soit la position de cette primitive dans la table de texture. La structure particulière utilisée pour le graphe médaillon permet de prendre en compte cette notion de quatre-voisinage pour toutes les primitives. Le calcul de la position des primitives est modélisé par un processus bidimensionnel dans lequel les contraintes relatives aux bords de la table de texture sont introduites. Les lois de probabilité a priori sur les distances et l'angle sont modifiées pour le calcul de la position de chaque nouvelle primitive.

La figure 3 illustre l'assemblage en parallélogramme utilisé. Les défauts apportés par la dimension finie de la table de texture sont visibles sur la photographie 1. Ces défauts sont absents sur la photographie 2 où la texture est obtenue avec la méthode exposée ci-dessus.

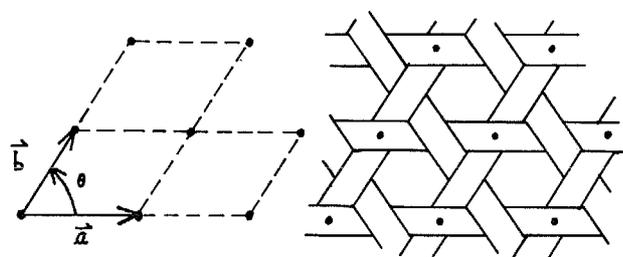


Figure 3 : assemblage en parallélogramme : graphe et exemple de texture

III - TRAITEMENT ANTI-MOIRAGE POUR LA VISUALISATION DES TEXTURES

La suppression des défauts qui apparaissent dans les textures à cause du sous-échantillonnage spatial au moment de l'affichage est obtenue par filtrage des hautes fréquences du signal texture. Norton et al [11] ont proposé un filtrage diminuant à la fois les composantes



SYNTHESE D'IMAGES AVEC TEXTURES STRUCTUREES ET EFFETS PARTICULIERS  
 IMAGE GENERATION WITH STRUCTURED TEXTURES AND SPECIAL EFFECTS

A. BRUNO , D. BARBA

hautes fréquences du signal (par suppression des termes d'ordre élevé dans le développement en série de Fourier du signal) et aussi des autres composantes lorsque le support du filtre est de taille importante. Feibush et al [12] ont proposé l'utilisation d'un filtrage spatial de la texture. Ils utilisent un filtre à support variable et à réponse impulsionnelle non-uniforme. Cette méthode exige en général la lecture d'un grand nombre de points texture et leur pondération pour le calcul d'un point d'image : elle est donc assez coûteuse. Williams [13] a montré récemment l'intérêt, du point de vue coût calcul, d'utiliser une table de texture contenant à la fois la texture originale et la texture filtrée. Les filtres qu'il a utilisés sont à support carré, de tailles linéaires successives d'un facteur deux. La détermination de la valeur d'un point image nécessite huit lectures de la table de texture, deux interpolations bilinéaires et une interpolation linéaire quelque soit la taille du support du filtre équivalent qui serait utilisé sur la texture originale. Le coût de cette méthode est donc assez faible et reste constant. Les résultats sont intéressants mais l'effet du filtrage réalisé est trop important lorsque le support du filtre n'est pas carré, d'où une perte de contraste et de détails visibles sur les facettes très inclinées.

La méthode que nous proposons consiste à utiliser une table contenant la texture originale et la texture filtrée avec tous les filtres possibles à support de taille variant également en puissance de deux pour remédier aux défauts existants dans la méthode de Williams. Le coût en calcul devient alors très faible puisqu'il suffit de deux lectures de la table de texture et d'une interpolation linéaire pour obtenir la valeur de la composante texture pour un point d'image. Ce coût est constant quelque soit la situation rencontrée et le filtrage reste d'une qualité supérieure à celui présenté dans la méthode précédente. La taille de la mémoire pour un jeu textures filtrées est seulement trois fois celui de la texture originale seule.

Les photographies 3 et 4 montrent la même texture sans filtrage et avec filtrage. La diminution du contraste de la texture devient sensible lorsque la fréquence spatiale de la texture approche la fréquence d'échantillonnage. Au delà de cette fréquence, le contraste devient nul.

#### IV - EFFETS PARTICULIERS : LA TRANSPARENCE, EFFET DE BRUME

Le but des effets particuliers que nous introduisons est d'améliorer le réalisme des images synthétiques. Une description de certains effets particuliers existe dans [1]. Avant de décrire les effets de transparence et de brume, nous présentons le modèle de calcul d'intensité lumineuse utilisé dans le cas où les facettes des objets sont opaques.

Bui TuongPhong [14] a le premier utilisé un modèle d'éclairage à plusieurs composantes : une composante pour la réflexion diffuse, une composante pour la réflexion ambiante et une troisième composante pour la réflexion spéculaire.

Dans le modèle que nous avons utilisé, seules les deux premières composantes existent car la composante de réflexion spéculaire donne un calcul beaucoup plus coûteux que les deux autres composantes et son utilisation n'est pas indispensable dans les types d'applications qui nous concernent.

Le modèle de base utilise une réflexion diffuse. Chaque surface possède un coefficient de réflexion diffuse  $R_p$  compris entre 0 et 1. A ce coefficient sont ajoutées deux valeurs donnant la teinte et la saturation de la facette. Trois coefficients de réflexion sont alors calculés pour passer de l'espace couleur

luminance-teinte-saturation à l'espace couleur formé des trois composantes Rouge-Vert-Bleu de l'image couleur. L'énergie lumineuse incidente au point P de la facette est obtenue par :

$$E_{Pd} = R_p \times \cos i \times I_d \quad (IV-1)$$

où  $i$  est l'angle entre la normale de la facette et la direction d'éclairage,  $I_d$  l'énergie émise par la source lumineuse (si  $i$  est supérieur à 90 degrés,  $E_{Pd}$  est alors nul). A  $E_{Pd}$  il faut ajouter une quantité  $E_{Pa}$  représentant l'énergie émise au point P due à la lumière ambiante dans la scène.  $E_{Pa}$  est donnée simplement par la relation :

$$E_{Pa} = R_p \times I_a \quad (IV-2)$$

où  $I_a$  est l'énergie lumineuse ambiante dans la scène. Les valeurs de  $I_a$  et  $I_d$  sont telles que leur somme est égale à 1. Au point P, l'énergie totale émise est donnée par :

$$E_p = E_{Pa} + E_{Pd} = R_p (I_a + \cos i \times I_d) \quad (IV-3)$$

La transparence ajoute à l'énergie issue du point P, l'énergie arrivant par l'arrière de la facette pour laquelle un coefficient de transmission  $T_p$  compris entre 0 et 1 est défini. Cette énergie  $E_{Pt}$  s'ajoute à l'énergie précédemment calculée, soit :

$$E_p = E_{Pa} + E_{Pd} + E_{Pt} \quad (IV-4)$$

En pratique, nous avons toujours  $T_p + R_p \leq 1$ . Un coefficient  $T_p$  est calculé pour chacune des trois composantes R-V-B. Le principal problème à résoudre est la connaissance de  $E'_{Pt}$ , c'est-à-dire de l'énergie lumineuse provenant de l'arrière de la facette en cours de synthèse. Une fois  $E'_{Pt}$  connue,  $E_{Pt}$  est obtenue simplement par :

$$E_{Pt} = E'_{Pt} \times T_p \quad (IV-5)$$

Les différentes méthodes pour calculer la grandeur  $E'_{Pt}$  sont plus ou moins liées aux méthodes d'élimination des facettes cachées.

Parmi les algorithmes les plus utilisés pour l'élimination des faces cachées, celui de Watkins [15] permet de traiter le problème de la transparence de façon complète, mais son coût en calcul reste élevé. L'algorithme de Warnock [16] est aussi bien adapté que le précédent mais sa mise en oeuvre est délicate. La méthode de synthèse qui utilise une mémoire de profondeur (ou Z-mémoire) est mal adaptée au problème et les erreurs sur la luminance sont inévitables (une étude des erreurs est faite en annexe 1). Les deux premiers algorithmes présentés travaillent dans l'espace image, leur coût augmente linéairement avec la complexité de la scène à afficher et de façon combinatoire avec la résolution de l'image. Les algorithmes qui suivent travaillent dans l'espace objet, leur coût est donc indépendant de la résolution écran, mais augmente de façon combinatoire avec le nombre de facettes ou d'objets présents dans la scène. Newell et al [17] classent les facettes par ordre de distances croissantes à l'observateur et commencent par engendrer les facettes les plus lointaines. Le problème du classement des facettes dans le cas général est complexe et le coût calcul peut être élevé. Cependant, cet algorithme peut être utilisé de façon très intéressante pour le classement d'objets polyédriques. Nous reviendrons plus loin sur cette possibilité qui suppose une base de données bien hiérarchisée pour être appliquée à des objets complexes. Weiler et Atherton [18] ont mis au point un algorithme permettant de traiter correctement l'effet de transparence, mais cet algorithme est très complexe et coûteux.

Le choix d'un algorithme résulte d'un compromis lié à différents éléments qui sont principalement : le coût calcul autorisé, la complexité de la scène, la résolution de l'image, la compatibilité avec la méthode utilisant la mémoire de profondeur pour l'élimination des



SYNTHESE D'IMAGES AVEC TEXTURES STRUCTUREES ET EFFETS PARTICULIERS  
IMAGE GENERATION WITH STRUCTURED TEXTURES AND SPECIAL EFFECTS

A. BRUNO , D. BARBA

faces cachées dans le cas des facettes opaques. Nous avons choisi d'utiliser l'algorithme de Newell, combiné avec la méthode de la mémoire de profondeur pour traiter l'effet de transparence. Dans notre base de données, tout objet possédant au moins une facette transparente possède un indicateur de transparence. Lorsqu'un objet transparent est sur le point d'être engendré, les facettes transparentes de cet objet sont mémorisées avec certaines caractéristiques géométriques concernant l'objet. Les facettes opaques sont affichées de façon classique en utilisant la méthode avec mémoire de profondeur. Pendant la synthèse des facettes opaques un processeur travaillant en parallèle, classe les objets, et pour chaque objet non convexe, classe les facettes les unes par rapport aux autres. Lorsque toutes les facettes opaques sont affichées et qu'il ne reste plus d'objets, les facettes transparentes sont affichées en commençant par les plus lointaines.

La détermination de la valeur correcte de  $E_p$  est ainsi réalisable et l'effet de transparence peut être appliqué sur de nombreuses facettes car le coût du classement des objets entre eux, éventuellement des sous-objets convexes ou des facettes entre eux lorsque les objets ne sont pas convexes, n'augmente jamais de façon combinatoire avec le nombre de facettes transparentes présentes dans la scène, mais avec le nombre d'objets présents.

La méthode qui vient d'être présentée permet un calcul exact de la luminance des facettes transparentes. L'effet présente un réalisme suffisant avec cependant le type de défaut suivant. Une sphère transparente a la même transparence au voisinage de ses bords qu'en son centre, ce qui est contraire à l'observation courante et peut nuire, dans certains cas, à l'interprétation de la scène. Les méthodes de lancer de rayons (ray-casting) [19] ne présentent pas ce défaut car le chemin du rayon lumineux est calculé à chaque fois qu'une nouvelle surface est rencontrée. Ceci permet de calculer l'atténuation supplémentaire qui résulte d'une plus grande traversée de matériau lorsque celui-ci est traversé dans une direction éloignée de la normale de ses faces. Les lois physiques [annexe 2] utilisées pour le calcul de la transmission du rayonnement par une épaisseur de matériau d'opacité donnée permettent de calculer la transmission d'une face lorsque sa normale et la direction du rayonnement font un angle non nul entre elles. Soit  $T_p(0)$  le coefficient de transmission sous incidence normale,  $T_p(\alpha)$  le coefficient de transmission sous incidence d'angle  $\alpha$ , alors  $T_p(\alpha)$  est donné par la relation :

$$T_p(\alpha) = T_p(0) \frac{I_\lambda(\alpha)}{I_\lambda(0)} = T_p(0) \times \exp[a \times L_0 \times (1 - \frac{1}{\cos \alpha})] \quad (IV-6)$$

Le calcul précédent est effectué une seule fois par facette transparente et n'est donc pas très coûteux, le terme  $\cos \alpha$  est connu puisque c'est la composante suivant la direction  $z$  de la normale de la facette. Le produit  $a \times L_0$  est directement donné par le coefficient de transmission suivant la relation suivante :

$$a \times L_0 = - \text{Log} [T_p(\alpha=0)] \quad (IV-7)$$

L'image 5 engendrée sans cette correction du coefficient de transmission montre que le réalisme obtenu est suffisant pour les objets qui sont à faces planes mais pose des problèmes d'interprétation pour les objets à surfaces courbes (sphère, cylindre). Avec la modification du coefficient de transmission, le réalisme présenté par l'image 6 est satisfaisant.

L'effet de brume consiste à considérer que le milieu dans lequel se propage les rayons lumineux est un milieu où le coefficient d'absorption  $\chi_\lambda$  [voir annexe 2] est constant pour  $\lambda$  donné (milieu homogène) et peut éventuellement varier avec  $\lambda$ , la longueur d'onde des rayons lumineux (brume plus opaque pour certaines

longueur d'onde ; application à la simulation de prises de vue visible-infra-rouge par exemple). La relation pour obtenir la luminance est :

$$L = L_0 \exp(-\chi_\lambda \cdot d) + H \cdot [1 - \exp(-\chi_\lambda \cdot d)] \quad (IV-8)$$

où  $L_0$  est la luminance propre de l'objet,  $H$  la luminance propre de la brume,  $d$  la distance observateur-facette et  $\chi_\lambda$  le coefficient d'atténuation de la brume.

Ce calcul doit être fait pour chaque point image sur les trois composantes couleur R-V-B, le coefficient d'absorption n'étant pas obligatoirement le même pour toutes les composantes, et la brume pouvant elle-même être colorée.

Les images 7 et 8 montrent un effet de brume appliqué sur une scène avec une densité de brume différente dans les deux cas. Le coefficient d'absorption est le même pour les trois composantes R-V-B. Cet effet donne un réalisme très intéressant dans le cas des simulateurs d'avions ou de bateaux.

Toutes les images présentées ont subi une correction non linéaire liée à la transformation non-linéaire entre amplitude du signal électronique et luminance sur l'écran de visualisation. La relation est du type :

$$L = I^\gamma \quad (IV-9)$$

avec  $\gamma$  voisin de 2.7 mais différent pour chaque composante R-V-B. Cette correction est effectuée par lecture d'une table précalculée pour chacune des trois composantes couleur R-V-B.

ANNEXE 1 : ETUDE DES ERREURS SUR LA LUMINANCE AVEC LA METHODE DE LA MEMOIRE DE PROFONDEUR

La méthode de la mémoire de profondeur rend inutile tout découpage entre les facettes et elle évite un classement des facettes en fonction de la distance observateur-facette. Le calcul de la profondeur  $z$  se faisant en parallèle avec celui des composantes couleur (R-V-B), cette méthode n'augmente que très peu le temps de calcul nécessaire pour un point image. En effet, c'est au moment de l'affichage du point image calculé que nous testons la profondeur  $z$  de ce point avec celui contenu dans la  $z$ -mémoire qui appartient à la dernière facette affichée.

Cette méthode est incomplète pour l'effet de transparence car même dans les situations les plus simples où seule une facette transparente est présente avec deux facettes opaques, alors suivant l'ordre dans lequel arrivent ces trois facettes, le résultat obtenu peut être très différent. La situation dans ce cas simple est développée ci-dessous, les facettes sont numérotées 1, 2 et 3 par ordre de distances croissantes à l'observateur. Il existe 6 cas avec un ordre d'arrivée des facettes différent.  $L^{(1)}$  représente la luminance et  $Z^{(1)}$  la distance à l'observateur suivant la direction  $z$  à l'étape  $i$ . La facette transparente est  $f_1$  ;  $f_2$  et  $f_3$  sont opaques.

CAS 1 : ORDRE D'ARRIVEE DES FACETTES =  $f_3, f_1, f_2$  .

$$L^{(1)} = L_3$$

$$Z^{(1)} = Z_3$$

$$L^{(2)} = ct_1 L^{(1)} + cr_1 L_1$$

$$Z^{(2)} = Z_1$$

$$L^{(3)} = L^{(2)}$$

$$Z^{(3)} = Z^{(2)} \quad \text{car} \quad Z_2 > Z_1$$

SYNTHESE D'IMAGES AVEC TEXTURES STRUCTUREES ET EFFETS PARTICULIERS  
 IMAGE GENERATION WITH STRUCTURED TEXTURES AND SPECIAL EFFECTS

A. BRUNO , C. BARBA

CAS 2 : ORDRE D'ARRIVEE DES FACETTES :  $f_3, f_2, f_1$

$$L^{(1)} = L_3$$

$$Z^{(1)} = Z_3$$

$$L^{(2)} = L_2$$

$$Z^{(2)} = Z_2$$

$$L^{(1)} = ct_1 L^{(2)} + cr_1 L_1$$

$$Z^{(1)} = Z_1$$

CAS 3 : ORDRE D'ARRIVEE DES FACETTES :  $f_2, f_3, f_1$

résultat identique au cas 2

CAS 4 : ORDRE D'ARRIVEE DES FACETTES :  $f_2, f_1, f_3$

résultat identique au cas 2

CAS 5 : ORDRE D'ARRIVEE DES FACETTES :  $f_1, f_2, f_3$

$$L^{(1)} = ct_1 L^{(0)} + cr_1 L_1$$

$$Z^{(1)} = Z_1$$

$$L^{(2)} = L^{(1)}$$

$$Z^{(2)} = Z^{(1)} \quad \text{car} \quad Z_2 > Z^{(1)} = Z_1$$

$$L^{(3)} = L^{(2)}$$

$$Z^{(3)} = Z^{(2)} \quad \text{car} \quad Z_3 > Z^{(2)} = Z_1$$

CAS 6 : ORDRE D'ARRIVEE DES FACETTES :  $f_1, f_3, f_2$

résultat identique au cas 5.

Le calcul exact est en fait donné par les cas 2, 3 et 4. Dans le cas 1, l'erreur vaut  $ct_1(L_3 - L_2)$ , dans les cas 5 et 6, elle vaut  $ct_1(L^{(0)} - L_2)$  avec  $L^{(0)}$  luminance initiale contenue dans la mémoire de luminance (en général zéro). Les erreurs sont donc très importantes.

En conclusion, il apparaît que même pour les cas les plus simples, la méthode de la mémoire de profondeur est inadaptée à l'effet de transparence sans adaptation très particulière.

ANNEXE 2 : CALCUL DE LA TRANSMISSION DU RAYONNEMENT LUMINEUX DANS UN MILIEU HOMOGENE ISOTROPE

Soit  $I_\lambda(o)$  l'intensité du rayonnement incident,  $I_\lambda(L)$  l'intensité du rayonnement transmis et  $X_\lambda$  le coefficient d'absorption sur la distance L. La relation entre  $I_\lambda(L)$  et  $I_\lambda(o)$  est donnée par :

$$I_\lambda(L) = I_\lambda(o) \cdot \exp(-X_\lambda \cdot L)$$

BIBLIOGRAPHIE

- [1] W.M. NEWMAN, R.F. SPROUL : "Principles of Interactive Computer Graphics" - 2<sup>nd</sup> édition, Mac Graw Hill, New York, 1979.
- [2] H. HÜGLI : "De la synthèse d'images appliquée aux maquettes de terrains numériques" - thèse de Doctorat, Ecole Polytechnique Fédérale de Zurich, 1980.

[3] H. TORRES : "Modélisation simple de textures aléatoires pour la synthèse d'images en temps réel" - Thèse de Docteur-Ingénieur, INSA Rennes, Nov. 1983.

[4] D. BARBA, A. BRUNO, H. TORRES : "Génération d'images synthétiques avec textures en temps réel" - Congrès IASTED, Lille, Mars 1983.

[5] R.W. EHRICH, J.P. FOITH : "A view of texture topology and texture description" - Computer Graphics and Image Processing, n°8, 1978.

[6] R. YOKOYAMA, R.M. HARALICK : "Texture synthesis using a growth model" - Computer Graphics and Image Processing, n°8, 1978.

[7] F.M. VILNROTTER : "Structural analysis of natural textures" - DPh, Univ. of Southern California, September 1981.

[8] S.W. ZUCKER : "Toward a model of textures" Computer Graphics and Image Processing, n°5, 1976.

[9] S.Y. LU, K.S. FU : "A syntactic approach to texture analysis", Computer Graphics and Image Processing n° 7, 1978.

[10] S.Y. LU, K.S. FU : "Stochastic tree grammar inference for texture synthesis and discrimination" - Computer Graphics and Image Processing, n°9, 1979.

[11] A. NORTON, A.P. ROCKWOOD, P.T. SKOLMOSKI : "Clamping a method of antialiasing for textured surfaces by bandwidth limiting in object space" - Computer Graphics, Vol. 16, July 1982.

[12] E.A. FEIBUSH, M. LEVAY, R.L. COOK : "Synthetic texturing using digital filters" - Computer Graphics, Vol. 14, July 1980.

[13] L. WILLIAMS : "Pyramidal parametrics" - Computer Graphics, Vol. 17, July 1983.

[14] BUI TUONG PHONG : "Illumination for Computer Generated Pictures" - CACM, June 1975.

[15] G.S. WATKINS : "A real time visible surface algorithm" - Univ. Utah Comput. Sci. Dept., June 1970.

[16] J.E. WARNOCK : "A hidden-surface algorithm for computer generated half-tone pictures" - Univ. Utah Comput. Sci. Dept., TR 4-15, 1969.

[17] M.E. NEWELL, R.G. NEWELL, T.L. SANCHI : "A new approach to the shaded picture problem" - Proc. ACM Nature Conf., 1972.

[18] K. WEILER, P. ATHERTON : "Hidden surface removal using polygon area sorting" - Computer Graphics, Vol. 11, n°2, 1977.

[19] E. GOLDSTEIN, R. NAGLE : "3D visual simulation" - Simulation, vol. 16, n°1, 1971.

REMERCIEMENTS

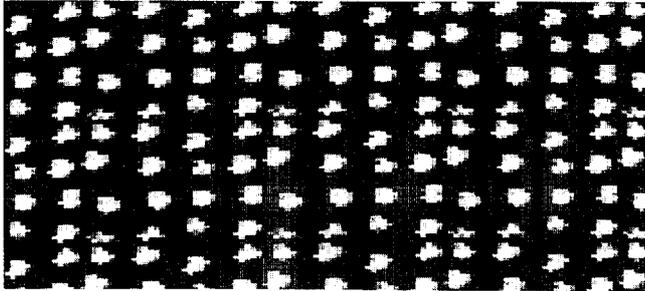
Cette étude est soutenue en partie par le CCETT (convention TDF - CCETT n° C 11 38 W) que nous remercions.



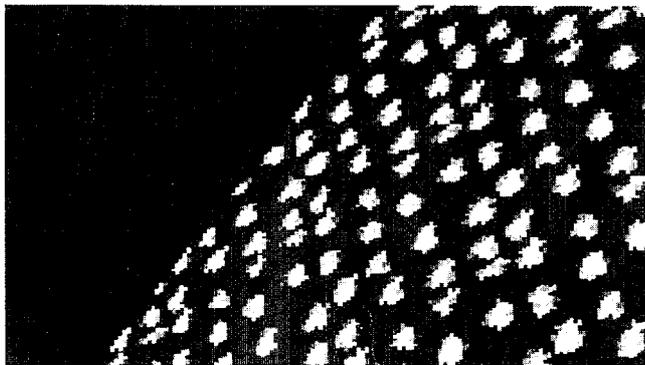
## SYNTHESE D'IMAGES AVEC TEXTURES STRUCTUREES ET EFFETS PARTICULIERS

## IMAGE GENERATION WITH STRUCTURED TEXTURES AND SPECIAL EFFECTS

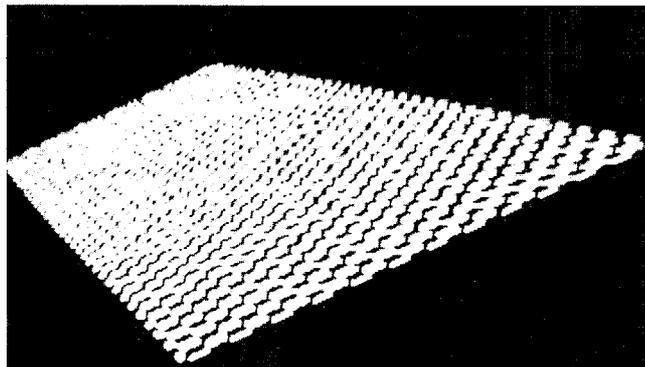
A. BRUNO , D. BARBA



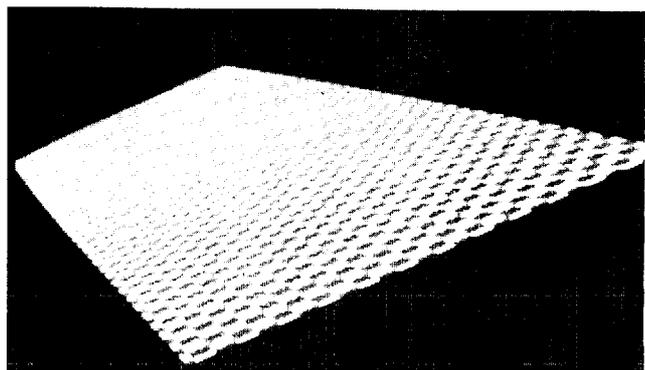
*Photographie 1 : texture engendrée sans contrainte pour les bords de la table.*



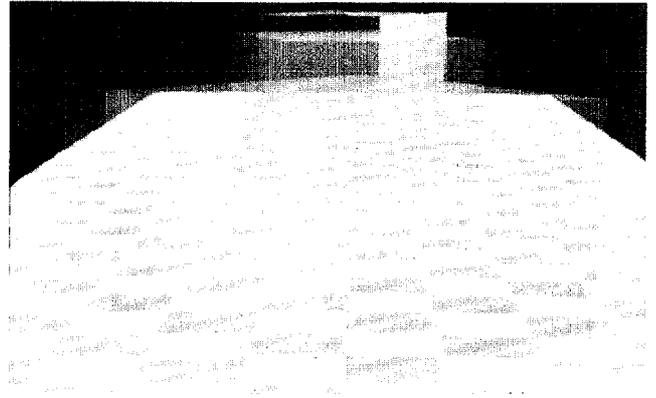
*Photographie 2 : texture engendrée avec contrainte pour les bords de la table.*



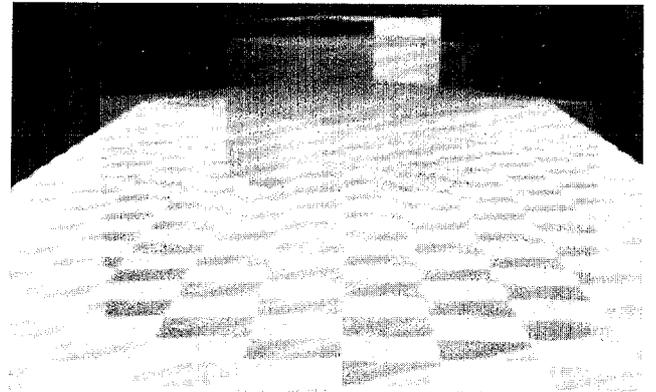
*Photographie 3 : texture sans filtrage.*



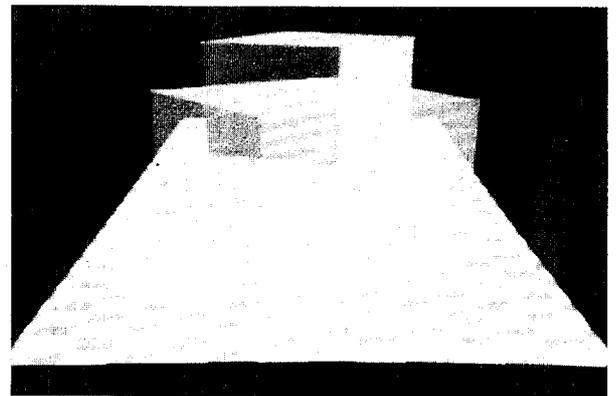
*Photographie 4 : texture avec filtrage.*



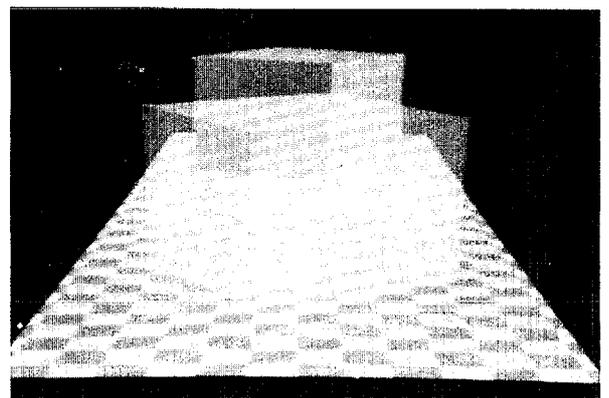
*Photographie 5 : effet de transparence sans correction du coefficient de transmission.*



*Photographie 6 : effet de transparence avec correction du coefficient de transmission.*



*Photographie 7 : effet de brume légère combiné avec transparence et filtrage.*



*Photographie 8 : effet de brume dense combiné avec transparence et filtrage.*