

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO INDUSTRIAL

Texturas em Síntese de Imagens

autor: Irene Andrea Velásquez Alegre 7/591
orientador: prof. Dr. Léo Pini Magalhães 7

Dissertação submetida à Faculdade de Engenharia Elétrica da Universidade Estadual de Campinas, para preenchimento dos pré-requisitos parciais para obtenção do Título de Mestre em Engenharia Elétrica.

Abril 1994.

Este exemplar corresponde à relação final da tese defendida por Irene Andrea Velásquez Alegre e aprovada pela Comissão Julgadora em 13.04.94.

Orientador

Resumo

Neste trabalho apresenta-se um estudo da Textura em ambientes de Síntese de Imagens. É apresentada uma revisão dos métodos mais conhecidos na modelagem e aplicação de texturas a objetos sintetizados por computador. Com base em experiências de texturização e no estudo do fenômeno físico por detrás da *textura observada* são propostos um Modelo Geral de Texturas e a especificação para um Módulo de Síntese de Texturas, no contexto do projeto ProSim. Finalmente, algumas imagens são apresentadas para evidenciar os resultados desta conceitualização. Pretende-se com este trabalho incrementar nossa experiência de texturização e fornecer as bases para o desenvolvimento de um sistema de textura no ProSim.

Abstract

This work presents a study of textures in Image Synthesis environments. A review of some well known methods for modeling textures and applying them to synthetic objects is presented. Based on our experience in texturing and the study of physical phenomena related to the *observed texture* a General Model for Textures and the specification for a Texture Module in the ProSim context are proposed. Some images are included to show the results obtained using this framework. This work intends to enhance our experience in texturing and provide the basis for development of a texture system in the context of project ProSim.

Agradecimentos

Agradeço a quantos deram apoio e incentivo para a realização desta dissertação, muito particularmente:

a Reginaldo, meu esposo, pelo amor, companherismo e o apoio efetivo na conclusão desta dissertação;

a Léo Pini, meu orientador pela paciência e incentivo;

aos meus pais, Juan e Delia pela dedicação e torcida silenciosa;

a Juan, Pepe, Perla e Nina, pelo amor e exemplo de esforço nas lutas de cada dia;

a Heraldo, pela sua disposição permanente de dar ajuda e compartilhar conhecimentos;

a Maurício França e Marcelo Malheiros pelas implementações do módulo de composição de padrões e dos algoritmos de Perlin, respectivamente;

a Luciana, pelas cenas criadas;

aos amigos Jorge, Inés, Silvinha, Trabuco, Andréia, Blanca, Gonzalo, Tânia Stahlke, Tânia Preto, Armando, Miriam, Sérgio, Adelaide;

à FAPESP, como órgão que financiou durante 30 meses este trabalho de pesquisa;

e principalmente, Àquele a Quem pertence toda honra, todo louvor, toda ciência, toda sabedoria e todas minhas ações de graças.

dedico esta tese a
Delia e Juan, meus pais
com amor e admiração.

Conteúdo

RESUMO	i
ABSTRACT	ii
AGRADECIMENTOS	iii
CONTEÚDO	i
LISTA DE FIGURAS	vi
LISTA DE TABELAS	viii
1 Introdução	1
1.1 Generalidades	1
1.2 O Projeto ProSim	2
1.3 Estrutura Geral	2
2 Métodos para Síntese de Texturas	4
2.1 Introdução	4
2.2 Mapeamento	4
2.2.1 Subdivisão da Superfície	5
2.2.2 Refletividade por Mapeamento	8

2.2.3	Simulação de Superfícies Rugosas	10
2.3	Texturas Estocásticas	14
2.3.1	Modelagem Estocástica da Textura	15
2.3.2	Movimento Browniano fracionário (fBm): Um Exemplo de Modelagem da Superfície Terrestre por Deslocamento de Pontos	17
2.3.3	fBm: Modelagem de Texturas Rugosas por Perturbação da Normal	20
2.4	Geração de Texturas a partir do Espectro de Frequência da Imagem	21
2.5	Modelagem de Texturas Naturais Usando Sistema de Partículas	25
2.6	Simulação de Propriedades Geométricas e Espectrais na Texturização	28
2.7	Textura Sólida	30
2.8	Texturização de Objetos Deformáveis	33
2.9	Hipertextura	34
2.10	Observações e Comentários	36
2.10.1	Fases na Síntese de Texturas	37
2.10.2	Uma Proposta Taxonômica	39
3	Um Estudo do Fenômeno Textura	41
3.1	Introdução	41
3.2	A Textura e o Fenômeno de Iluminação	42
3.2.1	A Luz	42
3.2.2	Reflexão, Transmissão e Absorção da Luz	44
3.2.3	Interferência, Difração e Polarização	46
3.2.4	A Cor	47
3.2.5	Fontes de Luz	50
3.3	A Textura e os Materiais	52
3.3.1	Índice de Refração	54
3.3.2	Efeitos do Tamanho da Partícula na Aparência do Material	55

3.3.3	Caracterização dos Materiais a partir dos Fenômenos de Reflexão, Transmissão e Absorção	56
3.4	A Textura e o Nosso Sistema de Visão	59
3.4.1	A Visão Humana - Aspectos Fisiológicos	59
3.4.2	O Modelo simplificado de Câmara Fotográfica	60
3.4.3	A Abordagem Psicologista de Percepção da Textura	61
3.5	<i>Rendering</i> e o Modelo de Iluminação em Síntese de Imagens	62
3.6	Resumo	65
3.6.1	Variáveis úteis para criar efeitos na Textura	66
3.6.2	Definições para a Síntese de Texturas	68
4	Uma Proposta para a Síntese de Texturas no ProSim	71
4.1	Introdução	71
4.2	Um Modelo Geral para Modelagem e Aplicação de Texturas em Ambientes Realísticos	71
4.2.1	A Síntese de Texturas	72
4.2.2	Espaço da Textura	73
4.2.3	A Função Parcial de Textura	74
4.2.4	O Mapa de Textura	75
4.2.5	A Função Completa de Textura	75
4.2.6	Funções de Aplicação de Textura	76
4.2.7	Relacionamento entre Funções Texturais	77
4.3	Estrutura para um Sistema de Síntese de Texturas no ProSim	79
4.3.1	O Módulo de Primitivas de Textura	81
4.3.2	Módulo de Ferramentas Gerais	81
4.3.3	O Módulo de Prototipação	82
4.3.4	Módulo de Funções para a Correspondência Espacial	83

4.3.5	Módulo de Aplicação da Textura	83
4.3.6	Módulo de Composição de Texturas	84
4.3.7	Módulo de Funções para Interação com o Usuário	85
4.4	O Processo de Texturização no ProSim	85
4.5	Exemplo: Como gerar uma textura realística de madeira	89
4.6	Observações e Comentários	90
5	Recursos para a Síntese de Texturas	92
5.1	Introdução	92
5.2	Fase de Modelagem da Textura: Algoritmos de Geração	93
5.2.1	Modelos Determinísticos	94
5.2.2	Modelos Estocásticos de Textura	97
5.2.3	Modelagem a partir da Transformada de Fourier	107
5.2.4	Composição de Texturas	107
5.3	Fase de Aplicação da Textura	114
5.3.1	Correspondência Espacial por Mapeamento	114
5.3.2	Correspondência Espacial por Textura Sólida	118
5.3.3	Funções de Aplicação	119
5.4	Observações e Comentários	121
6	Conclusões	122
6.1	Conclusões Gerais	122
6.2	Trabalhos Futuros	128
A	Algoritmos Geométricos para Geração de Textura:	131
A.1	Parâmetros Gerais:	131
A.2	Flags Gerais:	132
A.3	Primitivas de Textura e Mapas das Subregiões	132

Lista de Figuras

2.1	Subdivisão Recursiva do <i>Patch</i>	6
2.2	Algoritmo de Subdivisão Recursiva de Catmull	7
2.3	<i>Antialiasing</i> da textura por estruturas paramétricas piramidais	9
2.4	Vetor Normal Rotacionado	13
2.5	Textura com protuberância esticada	14
2.6	a) Perturbação na direção do eixo y - b) Perturbação da normal.	18
2.7	Ordem de subdivisão dos <i>patches</i>	19
2.8	Recomposição de um sinal através de soma de senoides	23
3.1	Reflexão, transmissão e absorção de um raio luminoso sobre um objeto. . .	45
3.2	Um objeto vermelho pelo princípio de absorção.	48
3.3	Onda luminosa resultante para fontes com diferentes distribuições espectrais de energia: (a) Luz solar; (b) Lâmpada incandescente.	51
3.4	(a) Superfície lisa; (b) Superfície semi-opaca; (c) Superfície opaca.	57
3.5	Escala de Texturização.	70
4.1	Interação das funções texturais no Modelo de Iluminação	73
4.2	Estrutura hierárquica da textura	78
4.3	Estrutura para um Sistema de Síntese de Texturas	80
4.4	Diagrama geral do Sistema de Texturas no ProSim atual.	80

4.5	Diagrama de blocos do Módulo de Prototipação	82
4.6	Fluxo de Procedimentos durante a texturização	87
4.7	Mapas de textura para a modelagem da textura “táboa de madeira” de esquerda a direita: Mapa para cor, mapa para o brilho, mapa para a rugosidade.	90
5.1	<i>t_flower</i> em diferentes instâncias para diferentes valores de cor, número de pétalas e tamanho da pétala	93
5.2	Pseudo-código da primitiva <i>t_circle</i>	94
5.3	Algumas das primitivas geométricas implementadas.	95
5.4	Diversos mapas gerados a partir da primitiva elipse.	96
5.5	Pontos vizinhos ao ponto $P(x, y, z)$	98
5.6	Instância da Função <i>t_turbulence</i>	102
5.7	Curva de Koch.	106
5.8	Matriz de modelos de textura de <i>diagonal_flower</i>	109
5.9	Cascata de Textura para <i>cascade_circle_flower</i>	109
5.10	Função <i>t_turbulence</i> composto com a primitiva <i>t_lista</i>	110
5.11	Repetições de uma primitiva <i>t_lista</i>	114
5.12	A mesma primitiva <i>t_lista</i> repetida em espelho.	115
5.13	Efeitos de variar a função de aplicação.	120

Lista de Tabelas

2.1	Uma proposta Taxonômica para Texturas.	40
3.1	A Cor absorvida e observada para vários comprimentos de onda.	48
3.2	Absorção da Luz Visível para Várias Cores	49
3.3	O efeito da cor da fonte de luz na cor resultante dos objetos.	50
3.4	Eventos moleculares induzidos por radiações de diversos comprimentos de onda.	53
4.1	Modelagem de características ópticas no modelo modificado de Whitted-Phong.	86
5.1	Resultados via Composição de Texturas.	111

Capítulo 1

Introdução

1.1 Generalidades

Em Síntese de Imagens o alvo na representação dos objetos é o realismo, e neste sentido a textura é uma característica muito importante pois acrescenta riqueza visual à superfície dos objetos modelados no computador.

Diversos aspectos do problema de texturização de objetos sintéticos têm sido tratados na literatura. No entanto as abordagens têm sido restritas a alguns poucos aspectos da textura que se deseja representar, existindo ainda vazios na conceitualização integral da textura e na diversidade de efeitos visuais relativos às mesmas.

A partir de uma abordagem psicologista e destacando o conceito de *textura observada* o presente trabalho tem por objetivo: desenvolver uma conceitualização geral para o problema de síntese de texturas de naturezas diversas, obter experiências pela implementação de técnicas de criação (modelagem) e aplicação de texturas nos objetos modelados no computador e definir uma estratégia para a implementação de um módulo de texturização dentro do projeto ProSim¹.

¹Prototipação e Síntese de Imagens e Animação - desenvolvido no Departamento de Engenharia de Computação e Automação Industrial da Faculdade de Engenharia Elétrica da UNICAMP

1.2 O Projeto ProSim

O projeto ProSim - Prototipação e Síntese de Imagens e Animação-, do Departamento de Engenharia de Computação e Automação Industrial da Faculdade de Engenharia Elétrica, é um projeto que vem sendo desenvolvido desde 1989.

O projeto possui vários módulos de desenvolvimento:

- modelagem de sólidos - nas modalidades de *b-rep (boundary representation)* e CSG (*Constructive Solid Geometry*),
- síntese de imagens - inclui algoritmos de *rendering*² para a visualização de objetos realistas nos módulos de *scan-line*, *ray-tracing* e radiosidade bem como o desenvolvimento de modelos de iluminação varios a serem usados nestes módulos,
- animação - incluindo módulos de animação cinemática e dinâmica.

Neste contexto o módulo de síntese de texturas é necessário como uma ferramenta para agregar realismo às cenas e cujos procedimentos serão chamados durante o *rendering*.

1.3 Estrutura Geral

Nos capítulos seguintes descreve-se o trabalho desenvolvido para alcançar os objetivos acima, segundo a estrutura a seguir.

O Capítulo 2 contém um compêndio das principais abordagens que na literatura tratam da texturização em ambientes de síntese de imagem. Também inclui algumas observações pretendendo organizar as diferentes abordagens e identificar as fases necessárias ao processo de texturização de objetos sintéticos.

No Capítulo 3 apresenta-se um estudo do “fenômeno textura” desde o ponto de vista da iluminação, do material e suas propriedades e do sistema de visão com o qual a textura é percebida. Também são discutidos o modelo para o transporte da energia luminosa

²processamento pelo qual é simulado o comportamento dos raios de luz num certo ambiente e feito o balanceamento da energia luminosa.

na síntese de objetos por computador, os modelos de iluminação aliados e a relação dos mesmos com a texturização de objetos sintéticos. A partir deste estudo é definida uma base conceitual para uma abordagem ampla do problema tratado.

No Capítulo 4 define-se um Modelo Geral para a síntese de texturas e apresenta-se uma Especificação Geral para o módulo de texturas no contexto do ProSim.

No Capítulo 5 descrevem-se detalhes de alguns dos algoritmos implementados para a modelagem do campo de textura, apresentando-se exemplos das imagens obtidas. Trata-se também da composição de padrões texturais como mais um recurso de geração de texturas.

No Capítulo 6 aparecem as conclusões e comentários finais ao trabalho realizado bem como sugestões para trabalhos futuros.

O Anexo A inclui uma descrição funcional das Primitivas de Textura.

Capítulo 2

Métodos para Síntese de Texturas

2.1 Introdução

Métodos para Síntese de Texturas

Este capítulo apresenta um resumo dos trabalhos mais relevantes existentes na literatura para a texturização de objetos em síntese de imagens. Este resumo é de importância para o embasamento teórico deste trabalho, sendo que algumas das técnicas estudadas foram selecionadas para sua implementação.

2.2 Mapeamento

O conceito de mapeamento, tão amplamente usado em síntese de imagens, foi introduzido por Catmull [Cat75] como uma técnica que permitia colocar texturas sobre a superfície dos objetos, obtendo-se como resultado um efeito análogo ao de colar sobre ela um papel aderente com propriedades elásticas.

Desde então, este conceito tem sido aplicado de diversas maneiras na simulação de texturas: mapeamento de cor, mapeamento de reflexão, mapeamento de rugosidade, etc. e tem-se feito extensões a este conceito também para espaços 3-D (Ver 2.7).

Com base no trabalho inicial de Catmull, o mapeamento pode ser definido como uma técnica que adiciona textura sobre as superfícies a partir do estabelecimento de uma

correspondência entre dois espaços: o espaço de definição do objeto (x, y, z) e o espaço de definição da textura (u, v) . A relação entre estes espaços nem sempre é trivial, pois os objetos possuem formas diferentes e, em muitos casos, a descrição das superfícies usando o sistema cartesiano de coordenadas aumenta a complexidade dos cálculos; além disto, é necessário que a textura a ser mapeada esteja definida em todo o domínio do objeto. Em termos práticos, na implementação do mapeamento, os objetos são definidos em coordenadas apropriadas sempre que possível (coordenadas esféricas, cilíndricas, polares, etc.) e a textura é definida num espaço paramétrico, onde as variáveis (u, v) tomam valores dentro do intervalo $[0, 1]$.

2.2.1 Subdivisão da Superfície

Uma vez definida a textura que se deseja colocar, como é possível relacioná-la ao objeto sintético? Na tentativa de adicionar textura e obter controle sobre os parâmetros de tonalização em cada ponto da superfície durante o *rendering*, Edwin Catmull propõe em 1975 [Cat75] o **Algoritmo de Subdivisão Recursiva** como uma solução à necessidade de subdividir a superfície dos objetos com custos mínimos de processamento, obtendo assim, o controle desejado no espaço discreto da tela. Este algoritmo marca um ponto importante na síntese de imagens pela sua simplicidade e custos computacionais mínimos.

Catmull estuda particularmente o caso de superfícies modeladas por *patches*¹, onde desconsiderando os aspectos de continuidade entre *patches*, aborda os problemas de modelagem, tonalização e texturização de superfícies curvas. Este estudo é dividido em 3 sub-problemas:

1. conseguir estabelecer correspondência entre os pontos da superfície e sua representação nos pontos discretos da tela;
2. remover as partes não visíveis do objeto que está sendo representado na tela;
3. determinar a intensidade a ser colocada em cada um dos pontos da tela.

No primeiro sub-problema, a correspondência é conseguida subdividindo recursivamente o *patch* até que cada *subpatch* cubra só um ponto de amostragem na tela (ver Figura 2.1). A

¹Pode-se traduzir como “retalho”, o qual é uma primitiva de modelamento de superfícies que representa uma porção de superfície usualmente delimitada por um pequeno número de lados.

partir de então, o ponto da superfície é representado por esse *subpatch* mínimo encontrado. Este processo é conhecido como o Algoritmo de Subdivisão Recursiva, o qual também é utilizado em outras técnicas de modelagem (além da modelagem por *patches*). Um esquema do Algoritmo de Subdivisão Recursiva aparece na Figura 2.2.

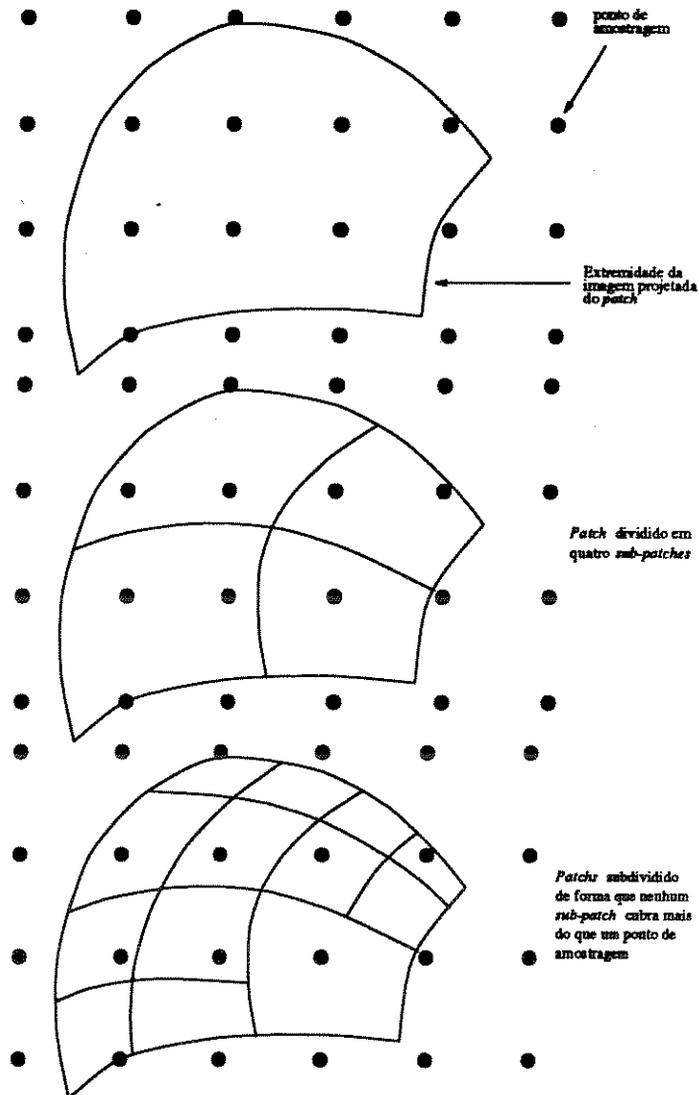


Figura 2.1:Subdivisão Recursiva do Patch.

O segundo sub-problema é abordado fazendo-se uso da técnica do *z-buffer*².

²Área de memória que contém a componente *z* da profundidade dos pontos numa cena.

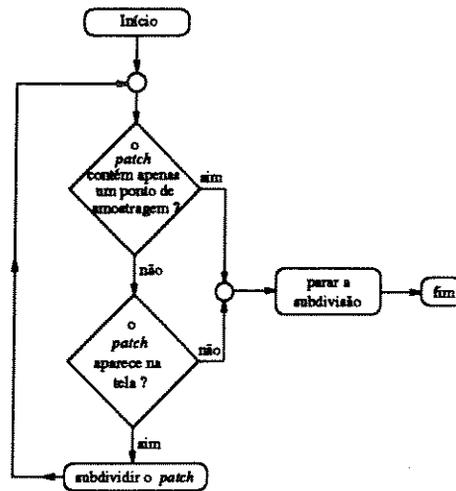


Figura 2.2: Algoritmo de Subdivisão Recursiva de Catmull

Quanto ao terceiro sub-problema, tendo-se já definida a unidade mínima de tratamento no espaço da tela (primeiro sub-problema) e sendo que o ponto vai aparecer visível na tela (segundo sub-problema), então, a intensidade I a ser considerada no ponto de amostragem da tela (*pixel*³) pode ser:

- uma função que varie segundo a normal à superfície do objeto que está sendo sintetizado: $I = f(normal)$;
- uma função que dependa da pressão, temperatura, altura, densidade do objeto no ponto em tratamento;
- o *bit-map*⁴ de uma imagem. Neste caso é necessária a definição de uma função que faça a correspondência entre o espaço de definição da imagem e o espaço do objeto. Esta função é conhecida como **função de mapeamento**: $I = f(u, v)$.

Desta maneira, se quisermos, por exemplo, texturizar um determinado *patch* com a imagem de uma fotografia, bastaria estabelecer a correspondência dos pontos $P(x, y)$ (pontos de amostragem da tela) com os $P(u, v)$ da fotografia.

³Do inglês "picture element" é a menor porção de memória endereçável do *frame-buffer*.

⁴Força-se o uso de *bit-map* para indicar a memória bi-dimensional usada para armazenar o valor de cada pixel de um arquivo de imagem

Um problema da proposta de Catmull é que não desenvolve efeitos mistos para superfícies texturizadas que devem ser tonalizadas, nem resolve os problemas de *aliasing*⁵, que podem ocorrer quando o número de pontos de amostragem é menor que o número de elementos armazenados da imagem; mas sugere o mapeamento de áreas no espaço da imagem sobre áreas no espaço da textura, como uma opção que não elimina, mas diminui consideravelmente o efeito de *aliasing*.

Catmull sugere também a possibilidade de usar o mapeamento para simular o efeito refletivo próprio de objetos com superfícies polidas ou espelhadas.

2.2.2 Refletividade por Mapeamento

Uma extensão às propostas de Catmull e uma solução para os problemas apontados ao final da seção anterior foi apresentada por Blinn e Newell em [BN76], onde exploram-se os efeitos de reflexões espelhadas no caso de superfícies muito polidas, a partir de técnicas da Teoria de Processamento Digital de Sinais, para diminuir o efeito de *aliasing* ao mapear texturas.

No que se refere ao Mapeamento de Texturas, Blinn implementa o efeito de filtragem sugerido em [Cro76]. Esta filtragem toma a forma de uma pirâmide, cuja base é a região 2×2 ao redor do ponto de amostragem da tela ou *pixel*, que está sendo inversamente mapeado ao seu quadrilátero correspondente no espaço paramétrico da textura. Aqui os valores das intensidades contidas no quadrilátero do padrão da textura são ponderados e somados à pirâmide distorcida que se forma ao encaixar a base quadricular do espaço da textura na base quadricular de 2×2 *pixels* na tela, tal como mostrado na Figura 2.3.

São também analisados os efeitos resultantes ao mapear sobre os objetos texturas geradas por técnicas diversas:

- a partir de algoritmos simples que implementam funções no espaço da textura (de coordenadas u, v);
- tomando imagens desenhadas a mão e digitalizadas como padrão de textura;

⁵Efeito indesejável que aparece quando sinais de alta frequência de uma imagem são representados como sinais de baixa frequência devido a uma amostragem inadequada do sinal.

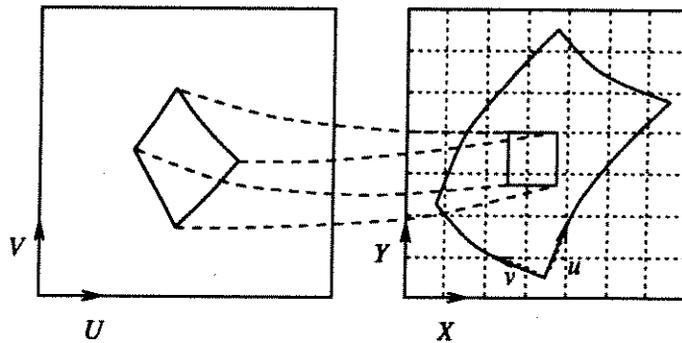


Figura 2.3: *Antialiasing* da textura por estruturas paramétricas piramidais

- utilizando fotos digitalizadas;
- com imagens geradas por Síntese de Fourier, obtidas por transformada inversa a partir do espectro de Fourier de alguma imagem bi-dimensional. Nesta técnica, sugere-se especial cuidado em modelar padrões contínuos nas bordas e cuja distribuição de intensidades seja mais ou menos uniforme a fim de não gerar deformações no mapeamento.

Em relação ao uso do mapeamento para criar o efeito espelhado que aparece sobre superfícies muito polidas, tal como sugerido por Catmull, estabelece-se em [BN76] que a simulação deste efeito precisa de uma modelagem cuidadosa das propriedades do objeto bem como de se poder determinar com precisão a posição da normal em cada ponto da superfície. Para adquirir esta informação, é usado o algoritmo de subdivisão recursiva. Através deste algoritmo é possível conhecer para cada ponto em tratamento: a direção da normal, a posição da fonte de luz e do observador, as regiões de superfície que possuem maior refletância-especularidade (para sobre elas simular os efeitos de reflexões de espelhamento). Um aspecto tratado refere-se a “o quê” vai ser refletido nestas regiões, neste caso Blinn e Newell propõem modelar o ambiente como uma projeção bi-dimensional do mesmo envolvendo o objeto. Por exemplo, simula-se que o objeto está posicionado no centro de uma grande esfera em cuja superfície interior é mapeado o ambiente. Assim, o ambiente é modelado como um mapa bi-dimensional que pode ser indexado pelos ângulos das coordenadas polares do raio refletido.

Ambas as técnicas, o mapeamento para dar texturas e o mapeamento para si-

mular reflexões espelhadas, podem ser combinadas para reproduzir superfícies texturizadas, que por serem muito polidas refletem os objetos ao seu redor.

2.2.3 Simulação de Superfícies Rugosas

Em relação à simulação de texturas, ficava ainda por representar o aspecto de rugosidade das superfícies. Em 1978 Blinn [Bli78] propõe uma técnica na qual, usando uma função de texturização, operam-se pequenas perturbações na direção da normal à superfície antes que esta seja usada para determinar a intensidade de cada ponto visível. Esta técnica consegue simular pequenas irregularidades que as superfícies reais apresentam, sem chegar a modificar realmente o perfil da superfície nas primitivas.

O desenvolvimento desta técnica é apresentado sobre superfícies modeladas por *patches*, mas é facilmente estendível a outras formas de modelamento. Na técnica desenvolvida Blinn considera uma superfície definida por *patches* paramétricos, onde:

$$X = X(u, v); \quad Y = Y(u, v); \quad Z = Z(u, v) \quad (2.1)$$

são funções bi-variantes. Então cada ponto genérico P da superfície pode ser representado pelo vetor:

$$\vec{P} = (X, Y, Z) \quad (2.2)$$

A normal no ponto P pode ser calculada como o produto vetorial de dois vetores tangentes à superfície nesse ponto:

$$\vec{N} = \vec{P}_u \times \vec{P}_v \quad (2.3)$$

onde:

$$\vec{P}_u = (\partial X/\partial u, \partial Y/\partial u, \partial Z/\partial u) \quad (2.4)$$

e:

$$\vec{P}_v = (\partial X/\partial v, \partial Y/\partial v, \partial Z/\partial v) \quad (2.5)$$

são as derivadas parciais de \vec{P} .

A idéia de perturbar a normal à superfície, em vez de modificar realmente o perfil da superfície, tem sustentação ao perceber-se que durante o *rendering*, a impressão

2.2 Mapeamento

de rugas superficiais relativamente pequenas deve-se principalmente ao efeito da variação da direção da normal à superfície e não à posição dos pontos da superfície em si.

Assim, uma perturbação qualquer da superfície de um objeto pode ser representada por uma função F , que somada à superfície original fornece o efeito de irregularidades nesta superfície.

Expressando matematicamente este raciocínio, suponhamos que a perturbação dos pontos P que descrevem a superfície (ver Eq. 2.2) possa ser representada pela função de perturbação F (em termos paramétricos: $F(u, v)$). Então, já que a perturbação é notória, principalmente devido aos desvios da normal, os pontos P' da superfície perturbada estariam definidos por:

$$\vec{P}' = \vec{P} + F \frac{\vec{N}}{|N|} = (X'(u, v), Y'(u, v), Z'(u, v)) \quad (2.6)$$

onde $\vec{N}/|N|$ é o vetor unitário que dá a direção da normal, \vec{P} define os pontos genéricos da superfície original e \vec{P}' representa os novos pontos, que resultam ao perturbar-se a superfície original por uma quantidade proporcional ao produto da função F na direção da normal.

Obtendo-se as derivadas parciais do novo ponto (P'_u e P'_v) e substituindo (2.6) nas equações (2.4) e (2.5) tem-se que:

$$\begin{aligned} \vec{P}'_u &= \frac{d}{du} \left(\vec{P} + F \frac{\vec{N}}{|N|} \right) = (\partial X'/\partial u, \partial Y'/\partial u, \partial Z'/\partial u) \\ \vec{P}'_u &= \vec{P}_u + F_u \frac{\vec{N}}{|N|} + F \left(\frac{\vec{N}}{|N|} \right)_u \end{aligned} \quad (2.7)$$

e

$$\begin{aligned} \vec{P}'_v &= \frac{d}{dv} \left(\vec{P} + F \frac{\vec{N}}{|N|} \right) = (\partial X'/\partial v, \partial Y'/\partial v, \partial Z'/\partial v) \\ \vec{P}'_v &= \vec{P}_v + F_v \frac{\vec{N}}{|N|} + F \left(\frac{\vec{N}}{|N|} \right)_v \end{aligned} \quad (2.8)$$

Simplificando, podemos dizer que, quando F representa pequenas rugas, esta é uma perturbação muito pequena e, portanto, o terceiro termo nas equações anteriores pode ser desprezado. Então as equações 2.7 e 2.8 ficariam:

$$\vec{P}'_u = \vec{P}_u + F_u \frac{\vec{N}}{|N|} \quad (2.9)$$

e

$$\vec{P}'_v = \vec{P}_v + F_v \frac{\vec{N}}{|N|} \quad (2.10)$$

A partir destas equações e, similarmente ao cálculo feito em (2.3), é fácil deduzir a expressão que define a normal ao novo ponto perturbado P' como:

$$\vec{N}' = \vec{P}'_u \times \vec{P}'_v \quad (2.11)$$

onde, substituindo-se 2.9 e 2.10 pode-se obter a nova normal:

$$\begin{aligned} N' &= \left(\vec{P}_u + F_u \frac{\vec{N}}{|N|} \right) \times \left(\vec{P}_v + F_v \frac{\vec{N}}{|N|} \right) \\ &= (\vec{P}_u \times \vec{P}_v) + F_u \frac{\vec{N} \times \vec{P}_u}{|N|} + F_v \frac{\vec{P}_v \times \vec{N}}{|N|} \\ &\quad + F_u F_v \frac{(\vec{N} \times \vec{N})}{|N|^2} \end{aligned} \quad (2.12)$$

Como o primeiro termo é por definição N e o último é zero (por tratar-se do produto vetorial de dois vetores paralelos), então, a equação da normal perturbada fica simplificada a:

$$\vec{N}' = \vec{N} + \vec{D} \quad (2.13)$$

onde

$$\vec{D} = \frac{F_u(\vec{N} \times \vec{P}_v) - F_v(\vec{N} \times \vec{P}_u)}{|N|} \quad (2.14)$$

Outro enfoque a este problema é também apresentado em [Bli78]. Neste caso, \vec{N}' é visto como o resultado de rotacionar \vec{N} de θ graus em relação ao eixo definido por: $\vec{N} \times \vec{N}'$. Então:

$$\vec{N} \times \vec{N}' = \vec{N} \times (\vec{N} + \vec{D}) = \vec{N} \times \vec{D} \quad (2.15)$$

que desenvolvendo fica

$$\vec{N} \times \vec{N}' = |N|(F_v \vec{P}_u - F_u \vec{P}_v) \equiv |N|\vec{A} \quad (2.16)$$

A partir de 2.15 e 2.16 ($\vec{N} \times \vec{D} = |N|\vec{A}$) e como D é perpendicular a N , então:

$$|D| = |A| \quad (2.17)$$

Como N , D e N' fazem um triângulo retângulo, então:

$$\tan \theta = \frac{|D|}{|N|} \quad (2.18)$$

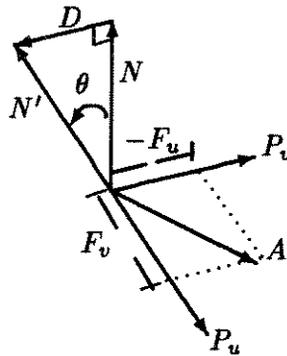


Figura 2.4: Vetor Normal Rotacionado

onde θ é o ângulo efetivo de rotação (ver Figura 2.4).

A função de perturbação F a ser usada para descrever o perfil das irregularidades da superfície, pode ser construída a partir de:

- o *bit-map* de alguma imagem (pode até ser o *bit-map* de um texto contendo letras);
- as informações do *z-buffer* derivadas da síntese de alguma imagem;
- imagens de desenhos computadorizados, com uma interpretação aliada de maneira a conseguir um determinado efeito. Por exemplo, que as áreas escuras produzam valores pequenos de F e as mais claras produzam valores grandes de F . Como são usadas apenas as derivadas de F , então áreas de intensidade constante (áreas homogêneas) apareceriam como áreas lisas na superfície resultante;
- um padrão gerado por algum algoritmo;
- um polinômio ou uma série de Fourier.

No caso de padrões que vão ser compostos, deve-se ter cautela ao criar desenhos que produzam continuidade nas bordas. Quando são usados polinômios descritos analiticamente ou por séries de Fourier, pode-se gerar um número excessivo de coeficientes, nesta situação, em que funções complexas devem ser descritas, podem ser usadas tabelas de dupla entrada, cujos valores possam ser acessados segundo as variáveis paramétricas u , v que definem a superfície.

Outro ponto importante é a dependência existente entre a textura rugosa e a escala do objeto. A escala do objeto pode variar se o observador for posicionado mais longe ou mais perto, porém tem-se observado que, quando isto acontece, a perturbação só afeta o objeto, enquanto que a função F não é escalada na mesma proporção (ver Figura 2.5). Para simular este efeito, uma função que faz variar a perturbação D à mesma taxa que N seria, intuitivamente:

$$D' = a \frac{D}{|D|} |N|, \quad \text{ou seja } D' = a|N| \quad (2.19)$$

pois $D/|D|$ representa a unidade.

Assim, temos que $a = \frac{D'}{|N|}$, o que faz lembrar (2.18). Portanto, igualando os termos, teríamos $\frac{D'}{|N|} = \tan \theta'$, equação esta que pode ser usada para criar perturbações dependentes da escala.

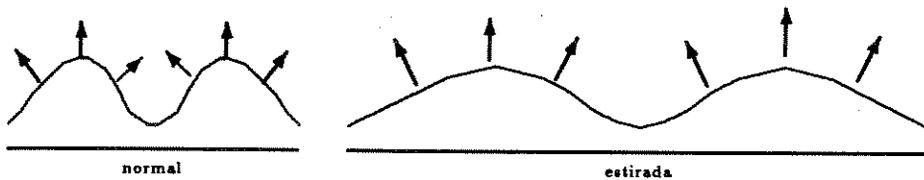


Figura 2.5: Textura com protuberância esticada

2.3 Texturas Estocásticas

Existem objetos ou fenômenos na natureza cujo comportamento é desordenado, por exemplo: o perfil do relevo da superfície terrestre, formações e movimento de fumaça, trajetória e movimento de um raio ou de uma folha em queda, contornos de mapas, um movimento turbulento, a superfície do mar, etc. Para estes fenômenos é usado o termo estocástico para indicar que a natureza de seu comportamento não é fixo ou previsível; porém assemelha-se à representação de processos estocásticos que contém variáveis aleatórias cujo comportamento é possível prever enquadrando-os dentro de uma ou mais das funções de distribuição de probabilidades bem conhecidas (distribuição gaussiana, de Poisson, de

Bernoulli, etc. [Pap91]).

Convencionalmente, texturas irregulares em pequena escala eram simuladas usando mapeamento com perturbação da normal [Bli78]; mas no caso destas irregularidades serem muito amplas, o algoritmo de Blinn mostra-se ineficiente (pois as irregularidades da superfície são simuladas por efeito de sombreado e não modificam o perfil do objeto) sendo preferível “modelar” as irregularidades da superfície.

Nesta seção são revisadas e estudadas algumas das técnicas para síntese de texturas estocásticas nas quais foi usado o Algoritmo de Subdivisão Recursiva de Catmull para a aplicação de texturas, estabelecendo uma correspondência entre o resultado de uma função estocástica e os pontos discretos da superfície na tela [FFC82], [HB84].

As vantagens da técnica de modelagem estocástica com relação às outras [Cat75], [Bli78] são:

- os objetos não precisam ser modelados em uma escala fixa;
- é possível representar objetos com relevo de detalhes irregulares na superfície.

2.3.1 Modelagem Estocástica da Textura

O objeto texturizado é modelado como a composição de uma primitiva determinística (que define a forma geral do objeto) e uma primitiva estocástica (que define a textura através da realização de algum processo estocástico, de uma ou mais variáveis). Deste modo o modelo poderia estar definido por uma equação paramétrica, tal como:

$$X(u, v) = P(u, v) + R(u, v, w) \quad (2.20)$$

onde:

u, v são variáveis paramétricas;

$P(u, v)$ é um polinômio que define a posição da superfície;

$R(u, v, w)$ é uma função f aleatória no espaço de amostras W com $w \in W$ e que causa variações estocásticas na posição da superfície através de um mapeamento do tipo:

$$f : \mathcal{R}^3 \rightarrow \mathcal{R}^2$$

$X(u, v)$ é uma função estocástica bi-dimensional que define a superfície.

Como $P(u, v)$ já faz parte de processos determinísticos bem conhecidos, o estudo das texturas estocásticas abrange somente métodos de geração de $R(u, v, w)$; os quais, a partir do contexto do algoritmo de subdivisão recursiva proposto por Catmull, podem ser aplicados no espaço discreto da tela durante o *rendering*.

O objeto desta modelagem não é a representação de uma instância particular do objeto (ou fenômeno, no caso de estar incluída a variável tempo) e sim a representação das características macroscópicas do objeto, de forma que sua “natureza” seja reconhecível, sem a necessidade do uso de uma grande base de dados.

Os elementos básicos para a modelagem estocástica são:

1. um objeto apropriado a ser modelado;
2. um processo estocástico apropriado que o modele;
3. um algoritmo que implemente realizações do processo estocástico para valores discretos na tela.

Considera-se um **objeto apropriado**, aquele com fortes e suficientes propriedades estocásticas de modo que possa ser alcançada uma boa representação do fenômeno e consideráveis economias em armazenagem e/ou tempo de processamento.

Um **processo estocástico apropriado** pode ser um modelo matemático do fenômeno a ser representado ou um processo estocástico escolhido empiricamente. Neste último caso, os valores dos parâmetros do processo estocástico devem ser bem definidos de maneira a representar com fidelidade o fenômeno em particular.

Um **bom algoritmo** para fazer a correspondência do espaço discreto na tela com os resultados do processo estocástico pode ser, por exemplo, a técnica de subdivisão recursiva proposta por Catmull [Cat75], pois:

- facilita o controle no detalhamento da textura, usando níveis de recursão e considerando a resolução da tela;
- possibilita o uso de fórmulas de interpolação;
- a textura retém suas características, mesmo quando vista a pequena distância, pois o *patch*, ou qualquer que seja a primitiva, pode ser subdividido a qualquer nível desejado;
- é necessário pouco esforço em processamento quando a textura é vista a grande distância, pois a subdivisão do *patch* é interrompida tão logo o tamanho do *patch* seja menor que o do *pixel*;
- os problemas de *aliasing* são resolvidos mesmo que o número de *pixels* ocupado pela textura seja muito pequeno comparado com o número de pontos de amostragem da função. A subdivisão pode continuar até o nível de *subpixels*; onde é eliminado o *aliasing*, ponderando as intensidades dos *subpixels*, até este nível.

Conceitos de modelagem estocástica foram usados por Fournier, Fusell e Carpenter em 1982 para representar irregularidades nas superfícies deslocando os pontos que as descrevem segundo a função estocástica. Haruyama & Barsky propõem uma representação similar, só que aplicando a função estocástica para perturbar a normal. Ambas abordagens são descritas a seguir.

2.3.2 Movimento Browniano fracionário (fBm): Um Exemplo de Modelagem da Superfície Terrestre por Deslocamento de Pontos

Em 1982 Fournier, Fusell e Carpenter [FFC82] propõem modular a superfície da terra através do movimento Browniano fracionário, pois a mesma é geralmente caracterizada por aspectos aleatórios, reconhecíveis a partir de certa distância de observação.

Como a modelagem estocástica proposta consiste em esculpir um relevo na superfície do objeto que está sendo texturizado, a perturbação obtida na implementação do fBm (movimento Browniano fracionário) é aplicada como vetores de deslocamento na posição espacial dos pontos que descrevem a componente determinística da superfície. Por

exemplo, no caso de uma superfície definida por *patches* e dado um sistema de coordenadas sobre o qual é definida a primitiva determinística, o resultado da perturbação (o deslocamento dos pontos) é aplicado sobre os pontos vértice que definem o *patch*.

Os vetores de deslocamento podem ser aplicados de duas formas: paralelamente à direção de um dos eixos coordenados no espaço objeto, como na Figura 2.6-a, ou na direção da normal (Figura 2.6-b), sendo que no segundo caso o resultado final é menos tendencioso.

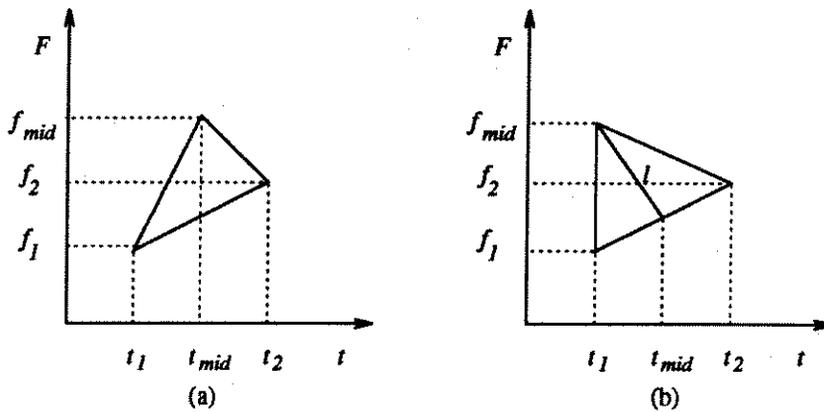


Figura 2.6:a) Perturbação na direção do eixo y - b) Perturbação da normal.

Quando este procedimento é aplicado a superfícies definidas por *patches*, como cada *patch* é calculado separadamente, não se pode dizer que a superfície como um todo possua as propriedades do fBm, nem que pontos coincidentes de *patches* vizinhos sejam necessariamente deslocados na mesma proporção. Neste caso, é necessário definir um $X(u, v)$ de tal forma que possa ser obtida uma função com uma aproximação razoável ao fBm e que ao mesmo tempo garanta as condições de continuidade interna e externa ao *patch*. Isto é possível introduzindo por exemplo uma interdependência entre os pontos calculados nas aproximações ao fBm, tal como mostrado no esquema da Figura 2.7. Nesta figura a ordem dos cálculos para os novos *subpatches* passa pelos níveis 0, 1a, 1b,..., onde, em cada nível, cada *subpatch* é tratado calculando-se os centros e logo os lados, usando-se em cada recursão os quatro vértices (diagonalmente para o centro, horizontal e verticalmente para os lados).

A cada passo o novo ponto é calculado como uma variável Gaussiana pseudo-aleatória cujo valor esperado é a média das vizinhanças no nível corrente e cujo desvio padrão é k^{-lh} , onde l é o nível corrente de recursão, h o parâmetro de auto-similaridade e

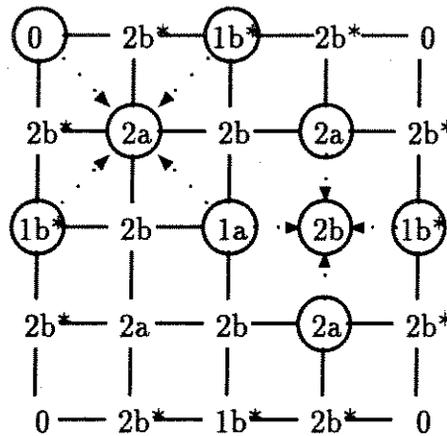


Figura 2.7: Ordem de subdivisão dos patches.

k uma constante de adequação à aplicação do modelo.

Em [FFC82] também é descrito o caso de primitivas representadas por funções paramétricas, pois esta abordagem conta com as seguintes vantagens:

- permite cálculos até níveis superiores ao píxel;
- só é necessário fazer cálculos para as áreas do *patch* que aparecem na tela;
- é fácil mudar o valor de h a qualquer nível de cálculo e com isto mudar propriedades do objeto estocástico (por exemplo, a rugosidade em função da distância do observador);
- no caso de se desejar dar propriedades estocásticas a objetos que necessariamente devem manter valores pré-determinados em certos pontos, é só forçar a que as componentes estocásticas sejam zero nestes pontos.

Finalmente, em [FFC82] sugere-se que da mesma maneira em que os resultados da realização dos processos estocásticos foram interpretados como vetores de deslocamento dos pontos que pertencem a uma superfície, também podem ser interpretados como perturbações de outras propriedades da superfície, por exemplo, cor, direção da normal (analogamente ao método de Blinn), etc.

2.3.3 fBm: Modelagem de Texturas Rugosas por Perturbação da Normal

Em 1984 Haruyama e Barsky [HB84] acolhem as sugestões de Fournier et. al e os resultados da implementação de modelos estocásticos para a representação de texturas rugosas, por alteração da normal.

Apresentam com detalhe como o método é aplicado para superfícies 1D e 2D, descritas por *patches* ou parametricamente, assim como duas possibilidades de introdução do fator aleatório: usando-o para alterar o ângulo definido pela normal e a tangente (neste caso o processo será mais custoso, pois inclui funções trigonométricas), ou como um vetor de deslocamento da normal.

Extensões à proposta inicial de Fournier et. al. são exploradas estudando-se o comportamento da imagem final, quando o parâmetro de auto-similaridade h cresce ou decresce, analisando-se também os efeitos da variação seletiva de h para determinados níveis de recursão.

Os resultados permitem identificar h como o responsável pela velocidade de crescimento ou decrescimento do fator estocástico, à medida que o nível de crescimento é mais profundo (taxa de crescimento = k^{-lh}). Assim, se h for 0, então o fator estocástico é o mesmo para todos os níveis de recursão. Um valor grande de h cria uma textura bastante lisa e um fator pequeno cria uma textura rugosa, isto significa que h tem uma relação inversa com o fenômeno “rugosidade” e ainda mais; com h pode-se controlar a distribuição espacial das frequências.

A idéia de que as variações de h podem exercer um controle preciso na distribuição espacial de frequências é implementada para modelar características texturais. É claro que o resultado não corresponde mais a uma superfície fBm e neste caso, a restrição de que h pertence ao intervalo $[0, 1]$ não precisa mais ser levada em conta, podendo h tomar inclusive valores negativos. Se h toma valores negativos então, a cada nível de recursão, a textura é ainda mais rugosa e o fator estocástico cresce rapidamente. Isto pode ser útil para criar texturas cada vez mais rugosas, a medida que são vistas mais de perto.

Como imitar uma textura natural Para imitar uma textura natural, Haruyama e Barsky acreditam que, se fosse possível calcular os valores de h baseados no espectro de

potência da textura, seria fácil criar uma textura similar sem empregar forma alguma de mapeamento, mas isto não é fácil. Uma análise em séries de Fourier também não funciona, pois esta se baseia nas premissas de amplitude, frequência e fase fixas. Portanto, o caminho mais rápido e factível seria o de se obter a textura por tentativa e erro, alterando-se os valores de h até obter resultados visualmente satisfatórios. Por exemplo, trabalhos experimentais mostram que é possível simular uma textura de laranja com o seguinte conjunto de valores: $h = 10$ no primeiro nível de recursão, com isto o desvio padrão fica pequeno para a próxima recursão, entre o 2º e o 6º nível $h = 0$, para o fator estocástico se manter constante e as perturbações pequenas e no 7º nível $h = -10$, com isto o desvio padrão cresce abruptamente por um fator de $2^{-(-10)} = 1024$, o que produz a aparição súbita de buracos. No 8º nível h é novamente 10, assim cria-se o efeito de saída do aprofundamento até a superfície lisa. Este método produz depressões igualmente espaçadas, cujo tamanho é o tamanho do *subpatch* no 7º nível de recursão. Esta textura, apropriadamente colorida consegue simular a superfície de uma laranja.

Este método não precisa de uma tabela contendo os dados de entrada que descrevam a função de perturbação, o que é uma vantagem em relação à perturbação da normal proposta por Blinn. Em geral pode-se conseguir texturas de complicado comportamento estocástico usando-se uma pequena quantidade de dados e sem a necessidade de ter o padrão da textura original como uma entrada ao sistema de criação de texturas.

2.4 Geração de Texturas a partir do Espectro de Frequência da Imagem

Uma característica importante das texturas naturais é que o detalhe textural acontece em mais de uma escala e este pode ainda variar segundo a escala em que está sendo visualizado. Por exemplo, uma área de pedras possui uma textura geral na qual identifica-se o contorno individual de cada pedra, mas que analisada ainda mais de perto mostra a textura da superfície de cada pedra em particular.

O modelo Browniano de Mandelbrot ou fBm, assim como outros processos que obedecem regras de formação fractal em geral, tem a propriedade de mostrar mais detalhes a medida que a distância de observação diminui, uma vez que possuem a propriedade

de auto-semelhança, segundo a qual os objetos são definidos como funções (analíticas ou probabilísticas) da escala.

Estes tipos de funções vistas no domínio da frequência correspondem às do tipo $1/f$, pois é possível caracterizá-las apenas com as frequências menores, mas se for requerido detalhe para escalas maiores as frequências mais altas serão incluídas.

Isto sugere que texturas deste tipo podem ser interpretadas e até modeladas, no domínio da frequência. Na abordagem de [FFC82] isto já é explorado e particularmente Haruyama e Barsky realizam experiências em que, para níveis de recursão mais profundos, os parâmetros da função que descreve a textura são modificados (ver 2.3.3).

Mas, em 1984 Lewis [Lew84] aponta que nem todas as texturas naturais possuem um espectro de frequência do tipo $1/f$ e apresenta um procedimento para produzir texturas aleatórias com espectro de frequência de comportamento arbitrário, a partir de um ambiente de pintura digital.

Para isto baseia-se no fato de que, ao considerarmos uma imagem por exemplo em 256 níveis de cinza, o gráfico do valor das intensidades correspondentes à sucessão de linhas da imagem, pode ser visto como um sinal de natureza ondulatória e portanto é possível a sua reconstrução a partir da somatoria de funções senoidais, onde o resultado final é uma boa aproximação do sinal original. A figura 2.8 mostra esta propriedade. Como as senoides componentes possuem frequências e fase diferentes, dizemos que o sinal original está agora sendo descrito no espaço das frequências. O procedimento pelo qual uma imagem é transformada do domínio espacial ao domínio das frequências é chamado de Transformada de Fourier e na literatura [Gon87] encontram-se procedimentos apropriados para a implementação da mesma a partir de uma função discreta.

Como a Transformada de Fourier pode possuir componentes complexas, quando se fala da “imagem do espectro” na verdade trata-se da representação em intensidades da parte real da transformada. Com base nesta teoria, Lewis propõe que em muitos casos é bem mais fácil “pintar” na tela do dispositivo de visualização a imagem de uma certa textura no domínio das frequências (e a partir daí gerar a textura desejada por transformação inversa) do que procurar algoritmos que representem diretamente a textura no domínio espacial.

Para Lewis, um procedimento de síntese de textura, a partir do espectro de

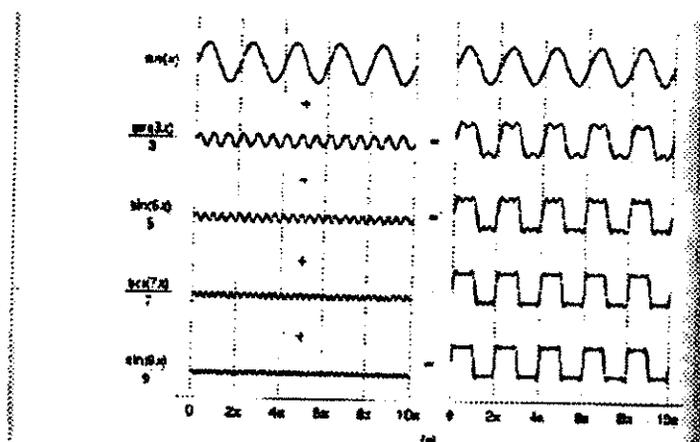


Figura 2.8:Recomposição de um sinal através de soma de senoides

freqüências, teria as seguintes etapas:

- i. o desenho do espectro de freqüências e a obtenção da textura desejada por transformação inversa;
 - ii. a convolução desordenada a partir da amostra obtida, a fim de gerar um campo mais extenso de textura;
 - iii. a filtragem do campo de textura para eliminar os ruídos introduzidos tendo como referência uma amostra da textura desejada.
- i) Nesta etapa, Lewis propõe que a imagem do espectro de freqüências necessária para a obtenção da textura pode ser diretamente criada através de um programa de desenho por computador ou de pintura digital; isto basicamente porque em muitos casos é muito mais fácil pintar o espectro de freqüências do que a textura final.

Para isto, e já que o espectro de freqüências tem propriedades de simetria, quando centrado nas freqüências de maior amplitude, propõe que para economizar esforços só seria necessário pintar ou desenhar dois quadrantes do espectro e o resto seria gerado por simetria. O espectro de fase poderia também ser gerado por uma função analítica, determinística ou aleatória.

Uma amostra da textura final seria obtida pela transformação inversa do espectro de frequências “pintado”, sendo que também a magnitude (espectro de potência), o logaritmo do espectro ou a fase podem servir como texturas.

ii) Lewis propõe a filtragem da amostra antes de sua repetição no espaço textural para eliminar as discontinuidades nas bordas, geradas durante o processo de transformação inversa, usando para isto a fórmula radial gaussiana $e^{-(x^2+y^2)^{1/2}}$.

O campo das texturas é inicializado num valor constante e desenvolvido como somas ponderadas da amostra em deslocamento. Os pesos e deslocamentos são dados por um gerador de ruído branco para que todos os sinais tenham a mesma probabilidade.

Este procedimento é equivalente a uma convolução desordenada ou a uma amostra da textura vista através de janelas que se deslocam com perturbação de ruído branco. Portanto a amostra textural assume o rol de resposta de impulso ao filtro ou *função de amplitude* e o espectro do campo de textura é o mesmo da amostra.

iii) Se um processo estocástico é modelado como um filtro de resposta a impulsos finitos excitado por ruído branco, então espera-se que uma amostra apropriadamente mascarada (filtrada) do processo contenha impulsos de resposta do filtro.

A transformada da área mascarada seria uma aproximação de banda-estreita à função δ , uma vez que a janela de modulação é equivalente à convolução do espectro da amostra com o espectro da janela.

Considerando no domínio espacial, a janela deveria ser grande o bastante para incluir um ou mais períodos da componente de menor frequência que caracteriza a textura. Para propósitos de pintura, ou síntese de imagens, pode-se definir o tamanho da janela por inspeção e ajuste.

Aplicação da Textura Como em [Lew84] a técnica de síntese de textura é desenvolvida a partir de um ambiente de pintura digital, uma vez desenhada a amostra textural, o usuário ou “pintor” deve descrever a extensão, a média e a dispersão com que suas “pinceladas” devem ser reproduzidas na geração do espaço textural. A textura final é obtida realizando mapeamento de cores sobre as intensidades do espaço textural e usando recursos diversos para interpolar, filtrar, modular, formar texturas como mosaicos repetitivos ou criar

sobreposição de texturas.

2.5 Modelagem de Texturas Naturais Usando Sistema de Partículas

Reeves em [Ree83] aborda a modelagem de certos tipos de objetos ou fenômenos naturais (fogo, água, núvens grama, etc...), como uma tarefa que deve levar em consideração a natureza difusa de seus alvos, uma vez que estes objetos não possuem superfícies lisas nem formas bem definidas e que, pelo contrário, são dinâmicos no tempo.

Isto representa uma nova forma de encarar as texturas de “objetos” que se encontram na natureza e que não são representáveis pelas primitivas convencionais (esfera, cone, cilindro, etc.), mas possuem uma estrutura de partículas organizadas num “sistema” cuja interação com a energia luminosa tem propriedades particulares [Bli78] e não pode ser caracterizada usando modelos convencionais de iluminação. Nesta situação a textura como uma impressão visual é o resultado de observar o fenômeno a uma maior ou uma menor distância e a modelagem da textura não pode ser isolada da modelagem do objeto ou do fenômeno em si.

Segundo Reeves, numa modelagem desta natureza considera-se:

- que o objeto não é representado por primitivas de superfície e sim como um conglomerado de primitivas de partículas que definem seu volume;
- o sistema de partículas é dinâmico pois a sua forma e posição espacial variam com o tempo;
- o objeto assim representado não é determinístico pois as mudanças em forma e aparência são governadas por processos estocásticos.

Geralmente, o modelo básico de um sistema de partículas é definido como um conjunto de muitas e diminutas partículas que juntas representam um objeto difuso. Num período de tempo, as partículas podem ser geradas, deslocadas no espaço, mudadas nas suas características (cor, forma etc) ou eliminadas do sistema.

Para definir um *frame*⁶ numa seqüência de movimento, são verificados os seguintes passos:

1. geração de novas partículas do sistema;
2. atribuição dos atributos individuais de cada partícula;
3. extinção das partículas;
4. transformações dos atributos dinâmicos das partículas;
5. *rendering* das partículas em atividade.

1) A geração de partículas é controlada por processos estocásticos. O número total de partículas no tempo pode ser, por exemplo, definida como:

$$n^{\text{ro}} \text{ de partículas} = \text{Quantidade Rand()} + \text{variância.}$$

2) Para cada nova partícula são definidos atributos variáveis e constantes, assim como valores para cada atributo, tais como:

- posição inicial;
- velocidade inicial;
- tamanho inicial;
- cor inicial;
- transparência inicial;
- forma;
- tempo de vida.

onde, por exemplo, a velocidade inicial pode estar definida como:

$$\text{veloc. inicial} = \text{veloc. média} + \text{Rand()} + \text{variância da velocidade.}$$

⁶Um *frame* representa uma imagem “congelada” em um determinado instante de tempo e que em conjunto com outros *frames* descrevem o movimento. Apesar da existência do termo *quadro* em português para o mesmo fim optou-se por esta expressão para evitar duplo sentido.

3) Uma partícula é eliminada quando excede seu tempo de vida (medido em *frames*) ou quando, pela trajetória de seu movimento, sai da janela de visualização do observador.

4) Apenas para aqueles atributos variáveis, uma fórmula geral pode ser:

$$\text{atributo final} = \text{atributo inicial} + f(t)$$

onde $f(t)$ é uma função aleatória que descreve o comportamento do atributo.

5) O *rendering* de um objeto definido por um sistema de partículas tem os mesmos problemas do que os de um objeto convencional, gerando superfícies escondidas quando algumas partículas estão diante de outras, quando existe transparência nas partículas, quando algumas partículas produzem sombra em outras, etc. Ainda mais, deve-se considerar casos em que objetos difusos coexistam com objetos determinísticos numa mesma cena.

Reeves menciona que podem ser definidas hierarquias de partículas, em que partículas sejam formadas por outras partículas, para que seja possível exercer controle sobre objetos difusos de comportamento mais complicado.

Quanto ao objeto como um todo, sua dinâmica ou aparência pode ser descrita por qualquer modelo computacional, por exemplo, pela solução iterativa de um sistema de equações parciais. Quanto aos procedimentos associados às variações de cada partícula, os autores usam realizações de processos estocásticos simples.

Vantagens: A representação por partículas, além de permitir representar objetos difusos ou de formas não determinísticas, apresenta outras vantagens com relação aos métodos convencionais de modelagem, a saber:

- as primitivas “partícula” são mais simples que *patches* ou volumes predefinidos;
- como a definição do modelo é procedural, o nível de complexidade durante a síntese está diretamente ligado ao número de parâmetros que se deseja controlar;
- o enfoque procedural diminui grandemente esforços humanos de modelagem da textura;
- os objetos assim modelados facilitam a representação de dinâmicas complexas, isto é especialmente útil para animação.

2.6 Simulação de Propriedades Geométricas e Espectrais na Texturização

Nas seções anteriores deste capítulo, a textura é vista como uma característica que é “alheia” ao objeto ao qual é adicionada durante o *rendering*. Dizemos “alheia” porque textura é relacionada com “formas ou desenhos” a serem colocadas na superfície dos objetos.

Em 1985 Carey e Greenberg observaram que superfícies aparentemente lisas apresentam rugosidade a nível microscópico (*microfacetas*), que as propriedades espectrais na extensão de uma superfície sofrem variabilidade quando os materiais não são homogêneos e que objetos de materiais diferentes dão impressões texturais diferentes aos olhos do observador mesmo tendo cores iguais. Com base nisto, em [CG85] a textura é estudada como o resultado da interação entre as características da superfície dos objetos e o comportamento da energia luminosa incidente, integrando ambos os aspectos. Em relação às superfícies, destacam as características macroscópicas e microscópicas da sua geometria. Em relação à interação com a energia luminosa, mencionam as propriedades espectrais da superfície.

A *geometria macroscópica* inclui características tais como rugas, protuberâncias, depressões, etc. Em geral, este conceito aplica-se a aspectos da superfície cujas dimensões sejam significativamente maior que o comprimento de onda da luz visível. Descritores macrogeométricos são a normal à superfície e as suas bordas ou contornos.

A *geometria microscópica* da superfície se refere às pequenas microfacetas que compõem a superfície, observáveis a nível microscópico. A altura e inclinação média destas microfacetas descrevem o grau de *rugosidade* da superfície. Em geral, *geometria microscópica* são as características da superfície cujas dimensões sejam comparáveis ao comprimento de onda da luz visível. A luz que incide sobre as microfacetas é refletida, absorvida ou transmitida segundo a microgeometria particular de cada superfície. Isto pode ser representado estatisticamente pela altura média das microfacetas e o seu desvio padrão.

As *propriedades espectrais* incluem características de absorção, reflexão e transmissão do material, não dependem das propriedades geométricas da superfície e sim do material e podem ser representadas como funções dependentes do comprimento de onda dos raios luminosos que atingem o material (espectro da superfície).

Carey e Greenberg propõem a simulação destes efeitos tratando:

- todos os parâmetros do modelo de iluminação como fatores sujeitos a variabilidade;
- estabelecendo numa base de dados de materiais:
 - as curvas espectrais para as propriedades de absorção, refletividade e transmissão do material;
 - os parâmetros estatísticos (média e variância) que simulam a distribuição microfacetada da superfície.

Simulação da Geometria Macroscópica Para simular aspectos da geometria macroscópica, podem ser aplicadas funções de perturbação sobre as primitivas da superfície, como na abordagem de Blinn para simular rugosidade perturbando a normal apresentada em 2.2.3, ou por deslocamento dos pontos da superfície, como na abordagem de Fournier et.al., vista em 2.3.

Simulação da Geometria Microscópica Modelar as milhões de microfacetas que compõem uma superfície implicaria em enormes gastos de armazenagem e tempo de processamento para os dados. Assim, outra possibilidade seria representá-la:

- por uma função de distribuição com uma média e variância;
- por funções que descrevem o comportamento dos raios refletidos como funções espectrais da superfície.

Na implementação de Carey e Greenberg, a função que descreve a distribuição dos raios refletidos *especularmente* pode ser desde uma simples função cosenoidal até funções exponenciais desenvolvidas dentro de modelos estocásticos. A distribuição representa a probabilidade de que uma faceta reflita especularmente a luz numa direção dada. No modelo de Phong (ver equação 3.1) o parâmetro n contém as atribuições de especularidade e é chamado de “*expoente especular da lustrosidade*”. Este expoente representa a “dispersão” da reflexão especular e modela implicitamente a distribuição das microfacetas da superfície. As superfícies lisas possuem um valor de N_s relativamente alto e as rugosas valores relativamente baixos.

A distribuição de raios refletidos especular e difusamente é modelada implicitamente através dos coeficientes K_s e K_d da superfície. Os autores estabelecem que a soma de ambos coeficientes deve ser a unidade ($K_d + K_s = 1$).

Os dois aspectos ressaltados acima (distribuição das microfacetas e curvas espectrais para refletividade da superfície) devem ser definidos e avaliados interrelacionadamente, pois em conjunto representam a rugosidade da superfície, observando-se que a medida que a componente especular cresce (K_s) o expoente especular (n) deve também crescer.

A implementação da perturbação dos coeficientes do modelo de iluminação pode ser feita de maneira similar à introduzida por Blinn para a perturbação da normal e o algoritmo de subdivisão recursiva também é útil nestes casos, dentro do conceito de mapeamento.

2.7 Textura Sólida

Em 1985 Perlin [Per85] e Peachey [Pea85] introduzem uma abordagem tridimensional ao processo de texturização que é chamada de **Textura Sólida**. Assim, a textura não é vista apenas como uma característica da superfície mas como uma propriedade intrinsecamente embutida nos materiais e que se manifesta aos olhos do observador através da superfície do objeto.

Funções que definem Texturas Sólidas A função de textura sólida para um dado parâmetro ρ de tonalização não é mais do que uma função definida para os pontos da superfície de um objeto em termos de suas coordenadas espaciais:

$$\rho = \rho(X, Y, Z)$$

No entanto, é bem mais fácil definir esta função no espaço tridimensional todo e não apenas nos pontos de uma determinada superfície. A textura sólida é aplicada usando as coordenadas espaciais como o argumento da função de textura. Para facilitar o uso da função de textura, o espaço 3D de definição da textura pode ser normalizado entre os valores $[0, 1]$ para cada coordenada X, Y, Z . Peachey sugere que para aplicações especiais, a função

de textura sólida pode ser avaliada usando como argumento outros parâmetros, tais como, a normal à superfície, a direção do observador, a direção das fontes de luz, etc.

Texturização de Superfícies Complexas Particularmente no mapeamento a texturização de superfícies compostas por *patches* ou primitivas diferentes poderia resultar em uma tarefa muito complexa se o que se pretende mapear for um único padrão sobre a superfície toda (devido à discontinuidade entre *patches*). Na utilização de uma textura sólida, todos os problemas devidos às relações entre os *patches* ou o sistema de coordenadas usado na definição de primitivas geométricas perdem relevância pois as funções de textura são independentes da geometria da superfície e do sistema de coordenadas em que esta foi definida. Uma vez que cada ponto numa superfície complexa ocupa uma única posição no espaço tridimensional, a função de textura pode ser diretamente avaliada nessas coordenadas sem precisar de informação adicional alguma.

Como Gerar Textura Sólida Todas as técnicas e métodos de geração de texturas bi-dimensionais são válidas para definir texturas sólidas quando estendidas ao domínio tridimensional. No entanto, se fizermos uma distinção entre as texturas obtidas por digitalização de imagens e as texturas criadas proceduralmente, as primeiras não são fáceis de captar exceto através de uma grande quantidade de seções transversais da textura 3D. Por isso, texturas geradas proceduralmente são preferíveis, pois:

- quando definidas em termos de parâmetros, pode-se controlar algumas características do material;
- podem ser diretamente avaliadas a partir do procedimento que as cria ou de tabelas criadas em uma pré-execução do procedimento;
- as necessidades de armazenagem são menores (por exemplo, no caso de uma imagem de resolução $512 \times 512 \times 512$ seriam necessários cerca de 134 milhões de bytes).

Alguns Modelos em Texturas Sólidas Modelos bi-dimensionais podem ser estendidos no espaço 3D. Por exemplo, esferas de raios variáveis dispersas aleatoriamente no espaço podem ser usadas para criar superfícies com protuberâncias, com buracos ou texturas em

que um material contém pigmentações circulares de uma outra cor ou material. Para obter modelos que tentem aproximar uma determinada textura, as variáveis podem ser restritas a um domínio de variação. Por exemplo, os raios das esferas e o espaçamento entre elas pode ser restrito a certos valores limites. Este modelo pode ser estendido considerando em vez de esferas formas mais complexas (elipsóides ou volumes de contornos perturbados) ou mistas.

A síntese de Fourier também é facilmente extensível ao caso 3D, por exemplo, para a simulação de mármore. Às ondas senoidais é atribuído volume, para serem vistas como veias de raio variável que atravessam um material. O ângulo de fase das funções senoidais são definidos aleatoriamente. O processo de crescimento destas “veias” é usado para gerar uma tabela de dimensão 128×3 . Um ponto da superfície é texturizado por interpolação bilinear dos oito vértices de um cubo imaginário que contém as coordenadas do ponto em referência.

Também podem ser usadas funções de projeção baseadas em texturas bi-dimensionais e projetadas ao espaço tri-dimensional. Assim, por exemplo, uma textura sólida poderia ser definida como:

$$\begin{cases} R(X, Y, Z) = \rho(X, Y), & \text{para } X, Y \text{ que pertencem a } [0, 1], \text{ e} \\ R(X, Y, Z) = 0, & \text{em outro caso.} \end{cases}$$

Cada elemento textural de ρ gera um paralelepípedo retangular que se estende infinitamente em ambas direções do eixo Z . Como uma extensão a esta técnica, pode-se dar um caminho para a projeção (por exemplo, uma trajetória ondulada).

Vantagens desta Técnica As principais vantagens desta técnica são:

- podem ser usadas todas as técnicas de modelagem de textura desenvolvidas para o espaço bi-dimensional;
- não são necessárias técnicas de mapeamento do espaço objeto ao espaço da texturas;
- portanto, o processo de texturização fica independente do sistema de coordenadas usado na definição do objeto, assim como da forma do objeto;
- objetos complexos são facilmente texturizados sem perda da continuidade da textura entre faces ou primitivas adjacentes.

2.8 Texturização de Objetos Deformáveis

Os modelos e técnicas de texturização descritas anteriormente são aplicáveis para simular texturas em objetos cuja forma é fixa. Mas suponha-se que se deseje representar um objeto maleável que possua textura, como por exemplo, uma massa de concreto cuja forma esteja sendo alterada. Neste caso as técnicas convencionais de mapeamento de textura podem ser utilizadas para simular as texturas de grão de areia e partículas de pedra, porém será necessário considerar o comportamento da textura em relação à natureza maleável do material em questão.

Em 1987 Wivill et al. [WBC87] representam os objetos maleáveis por superfícies de *isovalores* em um campo escalar definido por pontos chaves (do inglês *key points*) no espaço. Cada ponto chave é visto como uma fonte de energia, isto é, possui um valor de energia ou de campo e sua fronteira é uma superfície cujo valor de campo é constante. O objeto que corresponde a um ponto chave isolado é uma esfera. Vários pontos chaves produzem uma forma que é resultante da fusão das esferas correspondentes. Durante uma animação os pontos chaves podem mover-se uns em relação aos outros, causando uma mudança suave de forma.

Wivill et al. descrevem um método para estabelecer um mapa de textura que se refere apenas aos pontos chaves e não aos objetos. Cada ponto chave pertence ao seu espaço de textura abstrato. Assim, seja (x, y, z) um ponto no espaço definido em relação a um ponto chave localizado no centro de um sistema de coordenadas. O espaço abstrato de textura, é definido por:

$$f = F(x, y, z),$$

$$h = H(x, y, z),$$

$$c = C(x, y, z),$$

onde a tripla (f, h, c) é um ponto nesse espaço e as funções F, H, C são escolhidas livremente, conforme à aplicação.

A cada ponto P da superfície de um objeto maleável, deve-se atribuir uma tripla (f, h, c) . Isto é feito através de uma soma ponderada, em que cada ponto P_i contribui com

uma quantidade Q_i ao campo P , tal como colocado nas equações seguintes:

$$f = \frac{f_i Q_i}{\sum Q_i},$$

$$h = \frac{h_i Q_i}{\sum Q_i},$$

$$c = \frac{c_i Q_i}{\sum Q_i},$$

onde $\sum Q_i$ é o valor de campo da superfície.

Estabelecida a tripla (f, h, c) , pode-se então aplicar cor ou textura ao ponto fazendo uso de tabelas ou funções analíticas que proporcionam esta informação. Trabalha-se assim num espaço 3D $f - h - c$ como se fosse a aplicação de uma textura sólida a um objeto convencional.

2.9 Hipertextura

Embora os métodos descritos anteriormente pretendam caracterizar as mais variadas texturas, existem fenômenos naturais, tais como a pelagem animal, cabelo, etc. que seriam impossíveis de descrever por algum modelador de superfícies ou precisariam de processos certamente complexos para sua definição; ainda outros fenômenos, tais como fogo, nuvens ou fumaça, que nem sequer possuem contornos definidos.

Em 1989 Perlin e Hoffert [Per89] criam o termo *hipertextura* para caracterizar o fenômeno que existe na região intermediária entre objeto e não objeto e que consiste na representação visualmente realística de “forma+textura” através de uma nova técnica que modula a forma aplicando texturas procedurais a regiões volumétricas contínuas.

Este método é diferente das técnicas anteriores [Per85] porque permite manipular formas no espaço \mathcal{R}^3 e não apenas sobre a superfície dos objetos. A técnica, embora tenha sido desenvolvida empiricamente, produz resultados muito convincentes. Diferentemente do sistema de partículas proposto por Reeves, a modelagem de objetos difusos é tratada variando-se a densidade dos objetos, os quais são vistos como formados por uma parte sólida e outra difusa e não como um conglomerado de partículas.

2.9 Hipertextura

Desta forma a modelagem seria similar a de se trabalhar com um material sólido sobre o qual o detalhamento das características visuais da superfície é conseguida por ajustes finos, inserindo-se pontos de controle nas funções que descrevem a superfície. Para estes ajustes finos os autores desenvolveram um conjunto de ferramentas de modelagem, ou funções estocásticas, que possuem parâmetros de forma bem conhecidos e facilmente manipuláveis.

Modelagem da Hipertextura Cada função de distribuição de densidade com que é modelado um objeto é chamada de **Função Densidade do Objeto** ou $D(x)$, com domínio entre 0 e 1, que descreve a densidade do objeto através do espaço 3D. A região maleável do objeto consiste, nestes termos, em todos os x tal que: $0 < D(x) < 1$.

Sempre que se esteja modelando a região maleável do objeto as “ferramentas de modelagem” dão maior flexibilidade na criação das formas. Uma boa analogia é pensar nesta região como constituída por um material bastante maleável onde o usuário pode puxar, esticar, pressionar, torcer ou deformar, sempre de maneira controlável, os contornos do objeto.

As ferramentas de modelagem são conhecidas como **Funções de Modulação de Densidade (DMF)** ou f_i . Cada DMF é usada para controlar características espaciais, de forma que a hipertextura H é criada por aplicações sucessivas de funções de modulação de densidades sobre um objeto $D(x)$:

$$H(D(x), x) = f_n (...f_2(f_1(f_0(D(x))))))$$

As DMF básicas apresentadas em [Per89] são:

Bias onde $bias_b = t^{\frac{\ln b}{\ln 0.5}}$

Gain onde $gain_g$ é a spline de duas curvas $bias$ tal que:

se $t < 0.5$, então

$$bias_{1-g}(2t)/2$$

caso contrário

$$1 - bias_{1-g}(2 - 2t)/2$$

Noise com as características de [Per85]

Turbulence (idem)

e algumas outras funções aritméticas aliadas a operações de controle de fluxo.

Nesta modelagem também é definida uma generalização de operadores booleanos para combinar objetos com regiões hipertexturais, ao estilo da modelagem CSG. Estes operadores são: união, interseção, diferença e complemento.

Por exemplo, na modelagem de uma esfera centrada em c , com raio r e região hipertextural s , a função de densidade que a modela seria:

$$\begin{aligned}
 D_{[c,r,s]}(x) : \\
 r_1^2 &:= (r - s/2)^2 \\
 r_0^2 &:= (r + s/2)^2 \\
 r_x^2 &:= (x_x - c_x)^2 + (x_y - c_y)^2 + (x_z + c_z)^2 \\
 D(x) &:= \text{if } r_x^2 \leq r_1^2 \text{ then} \\
 &\quad 1.0 \\
 &\text{else} \\
 &\quad \text{if } r_x^2 \geq r_0^2 \text{ then} \\
 &\quad \quad 0.0 \\
 &\text{else} \\
 &\quad \frac{(r_0^2 - r_x^2)}{(r_0^2 - r_1^2)}
 \end{aligned}$$

Perlin e Hoffert acrescentam que, uma vez modelado o objeto, o processo de *rendering* deve considerar a natureza variável da densidade através do objeto e, portanto, ser capaz de avaliá-la através de seu volume através de um modelo de iluminação adequado, com o qual também se controlem propriedades como cor e opacidade do objeto, fazendo-se uso, por exemplo, de mapas de intensidades nas diferentes regiões.

2.10 Observações e Comentários

Neste capítulo apresentaram-se as abordagens mais importantes da literatura da área em relação à síntese de texturas em objetos realísticos. De uma análise do conteúdo da

bibliografia consultada, vê-se que, os mesmos termos são usados para significados diferentes. Por exemplo o termo “síntese de textura” é usado tanto para falar da modelagem do campo da textura [Lew84] [YH78] [KA84] [GM85] etc. quanto para indicar o relacionamento entre o espaço do objeto e o espaço da textura [WBC87] [Pea85] [SG86] ou para referir-se à interação luz-superfície [CG85] [Ama87] [Bli77] [Bli78] [FFC82].

Com o objetivo de deixar uma terminologia mais clara ao referir-nos à síntese de texturas para objetos realísticos, é feita uma distinção dos termos a serem usados nesta dissertação bem como uma caracterização das fases necessárias à síntese de texturas.

2.10.1 Fases na Síntese de Texturas

Percebe-se que em síntese de texturas duas são as necessidades básicas:

1. Como gerar ou modelar as texturas que se observam no mundo real, o que implica na geração de um campo de definição da textura . Isso será chamado de **modelagem do campo de textura**;
2. Como relacionar as texturas aos objetos modelados por computador. Isto chamaremos de **aplicação de texturas** ou **texturização** .

1. O **campo de textura** nesta abordagem é um termo abrangente que refere-se a qualquer amostra de textura gerada num espaço apropriado (espaço da textura ver 4.2.2) na memória do computador, pronta para ser aplicada a objetos sintéticos. Na modelagem do campo de textura identificam-se três fases:

- i. a descrição do campo textural;
- ii. a interpretação do campo textural.
- iii. a geração do campo textural;

i) A descrição do campo de textura, pode ser:

- gramatical;
- procedural;

- através de arquivos de dados ou tabelas que definem valores da textura em cada ponto do espaço;
 - através das medidas estatísticas da textura (média, variância e distribuição estatística);
 - no espaço das frequências;
 - formas híbridas das anteriores.
- ii) A interpretação da textura está relacionada a como é usada a informação contida na descrição a fim de gerar o campo textural. A interpretação portanto não precisa ser única para cada descrição, pois é na variedade de interpretações da descrição do campo textural que são geradas variedade de texturas finais.
- iii) Na geração do campo textural tem-se observado três tendências na literatura:
- (a) modelagem de texturas de comportamento bem ordenado, como por exemplo, texturas que são implementáveis com algoritmos geométricos;
 - (b) modelagem de texturas de comportamento desordenado e irregular, como por exemplo, as texturas achadas na natureza e que necessitam da inclusão do fator aleatório na sua implementação. Alguns bons resultados já foram obtidos com modelos estocásticos bem conhecidos (com o fBm), mas ainda são pesquisados modelos estocásticos para representar a grande maioria delas;
 - (c) Representações híbridas.

As três fases da modelagem existem de forma independente à técnica em uso. Por exemplo na abordagem de [Lew84] a descrição acontece quando modela-se a imagem do espectro através de alguma ferramenta de desenho computadorizado. A interpretação consiste no tratamento que é dado aos dados da descrição, por exemplo, a imagem pode ser interpretada como um comportamento no espaço das frequências, neste caso a geração do campo textural consistirá em aplicar a transformação inversa de Fourier. Uma outra interpretação pode ser tomar diretamente a imagem descrita como o campo textural alvo, neste caso a geração do campo textural fica confundindo-se com a geração da descrição. Embora Lewis chame à técnica proposta de "síntese de textura" estamos propondo que na verdade isto corresponde à mode-

lagem do campo textural, e que a síntese se completa quando o campo modelado é aplicado a algum objeto de uma cena sintética.

2. Quanto a **aplicação das texturas** nos objetos, dois são os aspectos relevantes:

- i. a **Correspondência Espacial** entre o espaço de definição do objeto sintético e o espaço de definição do campo textural;
- ii. a relação entre o campo textural e o objeto podemos também chamá-lo de interpretação na aplicação (esta idéia é desenvolvida em 4.2.6).

i) As técnicas usadas para a correspondência espacial são :

- O mapeamento bi-dimensional. O qual é equivalente a “pintar” a textura sobre as superfícies;
- a textura sólida como uma extensão do mapeamento bi-dimensional ao espaço tri-dimensional, o qual é equivalente a esculpir os objetos em um material qualquer, revelando-se a textura na superfície exposta ao observador;
- a texturização por modelagem de partículas, que é equivalente a “construir” a textura a partir de seus componentes estruturais.

ii) A interpretação na aplicação esta relacionada à interação luz-superfície e refere-se às características do objeto que o campo textural está descrevendo. Por exemplo, esta característica pode ser o comportamento da cor ou da rugosidade na superfície. Na literatura encontram-se termos como *color-mapping*, *bump-mapping*, *reflection-mapping*, etc. para referir-se a este relacionamento.

2.10.2 Uma Proposta Taxonômica

Uma proposta taxonômica dos trabalhos existentes na área é mostrada na tabela a seguir. Nesta a classificação foi feita considerando as fases de síntese de textura. As referências consideradas fazem parte dos artigos consultados como parte de nossa pesquisa bibliográfica embora nem todos eles tenham sido comentados nesta dissertação.

Tabela 2.1: Uma proposta Taxonômica para Texturas.

Fase de Modelagem do Campo Textural	
Critério	Foram Usados
Considerando Descrição	<ul style="list-style-type: none"> *Abordagem estrutural (com primitivas e regras para as primitivas) [DvO85] [PD84] [AS81] *Critério estatístico (em que medidas estatísticas são relacionadas a termos como <i>smooth</i>, <i>coarseness</i>, <i>gradient</i>) [GM85] [GM86] [AS81], [GM86] *Linguagem sintática (também chamada de gramatical) [FL79][Fu80][KA84] *Hierarquias em apenas dois níveis (macro-micro) [CG85] *Distorções de uma textura ideal *Abordagem espectral do campo textural [Lew84]
Considerando Geração	<ul style="list-style-type: none"> *A partir de amostras da textura que se deseja simular [GM85], [GM86] *Por algoritmos matemáticos: <ul style="list-style-type: none"> - com perturbação [Per85] [SA79] [Yes79] [SS91] - que implementam processos matemáticos (por exemplo iterações que descrevem um fractal)[Man75] [MV68] [FFC82] - que implementam processos estatísticos (p.e. cadeias de Markov)[HS80] [GM85] [GM86] - que simulam processos ou fenômenos naturais [FFC82] [Man75][MPC74],crescimento[YH78], evolução genética[Sim91], difusão por reação química[WK91] *Composição de elementos estruturais (padrões ou procedimentos) [SA79][AS81][MFV80][PD84]
Considerando Aparência do Campo Textural	<ul style="list-style-type: none"> *Texturas repetitivas (Ex. padrões) [SA79] [AS81] [MFV80] [PD84] *Texturas regulares (o comportamento é bem definido) [Pea85] *Texturas estocásticas (comportamento desordenado) [HB84] [Man75] [MV68] [FFC82] [MPC74] [WK91] *Hipertextura - entre a modelagem e a textura p. e. a simulação de pelos [Per89]
Fase de Aplicação da Textura	
Critério	Foram Usados
Considerando Processo Simulado	<ul style="list-style-type: none"> *Construção da Textura [Ree83] *Processos Mecânicos: <ul style="list-style-type: none"> Escultura da Textura [Per89], [Pea85] Pintura da Textura [Lew84]
Considerando Parâmetro de <i>Rendering</i> Afetado pela Textura	<ul style="list-style-type: none"> *Localização do Ponto: Displacement Mapping [Coo84] *Alteração da normal: bump-mapping [Bli78] *Propriedades espectrais: reflection-mapping e transparency-mapping [BN76][CG85]
Considerando o Relacionamento Espaço-objeto, Espaço-textura	<ul style="list-style-type: none"> *Mapeamento Bidimensional [Cat75], [BN76], [Cro76] *Mapeamento 3D - Textura Sólida [Per89], [Pea85] *Mapeamento pela Curva Geodésica [SG86]

Capítulo 3

Um Estudo do Fenômeno Textura

3.1 Introdução

A que se chama de textura? Às formas observáveis na superfície de um objeto? À impressão geral de um conglomerado de objetos numa área predeterminada? Ao resultado da interação matéria-energia luminosa? Ou ao que os nossos olhos vêem como resultado desta interação?

Estas e outras questões preliminares foram levantadas enquanto discorria-se como sintetizar texturas nos objetos. Percebeu-se então que para simular a textura era necessária uma melhor compreensão de “o que está acontecendo” quando vemos uma textura. Para isto e uma vez que a textura só é perceptível na presença da luz, que é necessário um objeto material que possua a característica textural e que é necessário um sistema de visão para perceber a textura, foram identificados três elementos necessários e complementares na presença de textura: Luz, Objeto e Sistema de Percepção Visual. Portanto, o nosso estudo tem sido relacionado a três abordagens: O estudo da textura a partir do ponto de vista do fenômeno de iluminação, o estudo da textura considerando as características do material e os processos de transformação nele efetuados, o estudo da textura considerando-a dependente das virtudes e limitações do nosso sistema de percepção visual.

Neste capítulo apresenta-se um resumo dos tópicos estudados dos que interessou-nos particularmente: evidenciar o relacionamento do tópico em estudo com a impressão textural ou caracterizar seu efeito na textura; destacar que existem elementos variáveis no

modelo do fenômeno físico, úteis para a modelagem da textura; apresentar e discutir o processo de *rendering* como o procedimento no qual todos os fatores variáveis estudados são organizados e relacionados para a simulação da interação luz-material-sistema de visão e finalmente definir o escopo de alguns conceitos como a base para propor um modelo geral para a síntese de texturas a ser apresentado no capítulo 4.

3.2 A Textura e o Fenômeno de Iluminação

A forma e características da superfície dos objetos só são distinguíveis quando existe luz incidindo sobre eles. O nosso sistema de visão permite que o comportamento na superfície dos objetos de estímulos correspondentes às características ópticas de cor, brilho, opacidade, transparência, rugosidade, etc., possam ser identificados e processados em nosso cérebro para poder “reconhecer” determinada textura. Mas como aparece o brilho, opacidade, a cor dos objetos? Para explicá-los precisamos entender o fenômeno chamado “iluminação”.

3.2.1 A Luz

A iluminação acontece na presença de luz, a qual tem sido definida como “o aspecto visual da energia radiante”; uma definição mais explícita é: “o aspecto da energia radiante que o observador humano pode perceber através das sensações visuais decorrentes da estimulação da retina dos olhos” [Con58].

Convencionalmente [Min54] [WB72] [Bri80] a luz é explicada em parte pela teoria das ondas eletromagnéticas formulada por Maxwell e em parte pelos conceitos de fótons e relatividade formulados por Einstein. Finalmente, a mecânica quântica aparece para unificar estas naturezas onda-partícula aparentemente incompatíveis.

Do ponto de vista da teoria eletromagnética, a luz pode ser estudada como uma onda de deslocamento retilíneo que não precisa de um meio material (água, ar, etc.) para se deslocar, possui um espectro de radiação entre [400-700] nm. e se move livremente no espaço a 300.000 km/seg, embora a sua velocidade e direção possam variar ao penetrar os materiais. Em 1672 Newton formula sua teoria da cor observando que a luz branca de-

3.2 A Textura e o Fenômeno de Iluminação

compõe-se em várias cores. Na verdade, na luz branca ondas de energia luminosa de todos os comprimentos de onda viajam à mesma velocidade no espaço vazio, mas em outros meios viajam a velocidades diferentes, separando-se segundo seus diferentes comprimentos de onda (vermelho, laranja, amarelo, verde, azul, violeta), fato este conhecido como **dispersão**. Segundo a teoria de Einstein, a luz comporta-se como uma torrente de partículas chamadas fótons, cuja propriedade mais evidente é a energia a eles associada. A natureza particulada da luz se faz evidente pelo efeito foto-químico que acontece em materiais sensíveis (como um filme fotográfico) quando expostos à energia radiante; nestes acontecem reações foto-químicas como se o material tivesse sido bombardeado por um jato de partículas nas áreas atingidas pela luz. Estas reações explicam-se pelas mudanças nos níveis de energia das partículas de um átomo estável (por exemplo, os da substância fotoquímica na superfície do filme fotográfico) quando colidem com outras (por exemplo, os fótons contidos num raio de luz), excitando-o. O átomo logo retorna a seu estado de equilíbrio, liberando nesse processo energia luminosa numa ou mais etapas. A energia associada aos fótons torna-se evidente observando o relacionamento entre as equações

$$\lambda = C/f$$

e

$$E = hf$$

onde:

λ = comprimento de onda (nm)

f = frequência da onda (ciclos/seg)

C = velocidade da luz no vácuo (3×10^8 m/seg)

E = energia (Joules)

h = constante de Planck (6.63×10^{-34} Joule-seg)

Ambas equações expressam o mesmo fenômeno, a saber que associado ao fóton (modelados como ondas caracterizadas por λ e f) existe um nível de energia com que o fóton vibra e portanto quando impacta a nossa retina, o estímulo causado relativo a esta energia é interpretado pelo nosso cérebro como uma sensação de cor. Desta forma estabelece-se

o relacionamento comprimento_de_onda-energia-cor (ver Tabela 3.1 - última coluna). Os níveis mais energéticos produzem luz azul e os menos energéticos luz vermelha, a luz branca deste ponto de vista é uma mistura de fótons com todos os níveis de energia.

Devido à dupla natureza da luz existem duas perspectivas que podem ser usadas para estudá-la. Na perspectiva macroscópica, a região em estudo é muito grande comparada ao tamanho de átomos e moléculas. Esta perspectiva é usada no estudo das ondas e sua propagação em meios considerados contínuos. A segunda perspectiva, chamada microscópica, considera uma região muito menor e lida com os encontros entre as ondas ou fótons e átomos ou elétrons individualmente.

3.2.2 Reflexão, Transmissão e Absorção da Luz

A partir da perspectiva macroscópica, sabe-se que, quando a energia radiante de comprimento de onda λ encontra-se com a superfície dos objetos, parte é refletida ($\rho(\lambda)$), parte absorvida ($\alpha(\lambda)$) e parte transmitida ($\tau(\lambda)$). Como a energia deve ser conservada, então:

$$\alpha(\lambda) + \rho(\lambda) + \tau(\lambda) = 1$$

Quanto à parte refletida, se os raios refletidos são concentrados numa mesma direção dizemos que a **reflexão é especular**, neste caso os raios que abandonam a superfície fazem-no com um ângulo aproximadamente igual (em relação a normal à superfície) ao dos raios incidentes. Dizemos que o ângulo é aproximadamente igual pois tal como provado em [TS67] podem acontecer pequenos desvios quando a superfície não é um perfeito espelho, porém possui irregularidades que por serem tão pequenas produzem um tipo de reflexão que é ainda considerada especular. Quando os raios refletidos dispersam-se em todas as direções, como resultado da geometria randômica irregular da superfície, ou retornam de dentro do material por dispersão (isto acontece em materiais translúcidos não homogêneos), a **reflexão é difusa**. À medida que a superfície é mais rugosa a quantidade de reflexão especular diminui, em contrapartida, a quantidade de reflexão difusa aumenta. A transmissão pode ocorrer com ou sem dispersão e acontece quando o raio luminoso atravessa o material. A

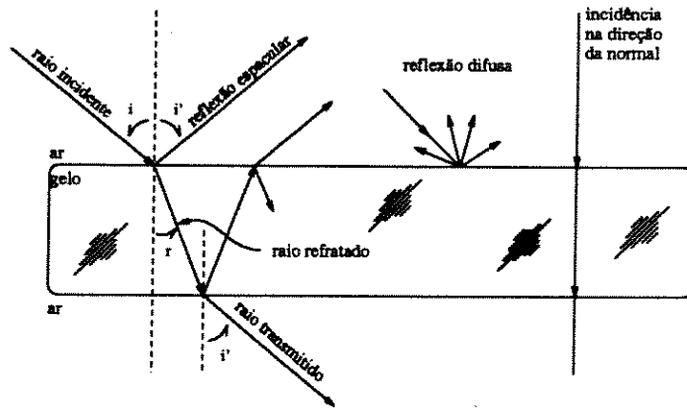


Figura 3.1: Reflexão, transmissão e absorção de um raio luminoso sobre um objeto.

Figura 3.1 mostra o comportamento do raio luminoso ao incidir sobre a superfície de um objeto.

O mecanismo é sem perda se $\rho(\lambda)$ e $\tau(\lambda)$ são as frações totais refletidas e transmitidas, respectivamente, em todas as possíveis direções (não acontece absorção de luz pelo material); $\rho(\lambda)$ é chamada de refletância espectral e $\tau(\lambda)$ é a transmitância espectral.

O comportamento dos raios luminosos devido a $\alpha(\lambda)$ e $\tau(\lambda)$ vão determinar entre outras coisas a cor do objeto e a determinação destas variáveis introduz a necessidade de um estudo particular para cada material.

Neste trabalho, o fato de se analisar o percurso do raio de luz fora do material não significa que a emissão, reflexão, absorção e transmissão de energia radiante sejam fenômenos da superfície, pois a emissão é a transformação da energia molecular ou atômica em energia radiante e para que a radiação tenha sido causada esta deve tomar lugar dentro do meio abaixo da superfície, pois tal como se lê em [WB72] pág. 31: “é necessário que aconteça alguma penetração de energia radiante ao meio abaixo da superfície e mesmo a reflexão na superfície dos metais envolve penetração, embora seja esta bastante superficial, a uma profundidade finita do metal”. Mesmo assim, por razões de simplicidade estes fenômenos podem ser tratados como se acontecessem essencialmente na superfície.

3.2.3 Interferência, Difração e Polarização

Além dos efeitos ligados à reflexão, transmissão e absorção, as vezes é possível observar outros efeitos associados aos fenômenos conhecidos como interferência, difração e polarização. Estes fenômenos são observados menos frequentemente e se originam devido à natureza ondulatória do raio luminoso. Para explicá-los vale lembrar que o modelamento material do raio luminoso tem tomado convencionalmente duas formas: a de uma linha reta que descreve o caminho percorrido pelo fóton ou a de uma onda geralmente do tipo senoidal. A primeira destas formas é usada na chamada “óptica geométrica”, simplifica bastante o estudo do comportamento do raio luminoso pois permite que cada raio possa ser tratado independentemente e consegue aproximar a maioria dos efeitos visuais que se observam no mundo real (esta abordagem é a mais comum em computação gráfica). A segunda forma, ou seja a modelagem do raio luminoso como uma “onda” é menos usada devido à representação mais custosa dos efeitos visuais mais comuns porém é uma representação necessária para gerar os efeitos de interferência, difração e polarização.

A interferência é observada entre ondas luminosas que ao se sobreporem geram modificações entre elas de maneira que as ondas se somam construtiva ou destrutivamente, sendo que o ângulo de fase entre as mesmas é fator determinante para a ocorrência de um ou outro caso. Exemplos visíveis de interferência são observados nos anéis coloridos que se formam sobre superfícies oleosas, sobre bolhas de sabão ou mais impressionantemente nas coloridas penas do pavão e na superfície da madrepérola. Uma particularidade deste efeito é que as cores não parecem ficar estaticas mas variam quando o observador muda sua posição.

A difração acontece nas pequenas mudanças de direção que o raio luminoso faz ao evitar uma certa partícula (a magnitude do desvio depende do comprimento de onda do raio e do tamanho da partícula). Estes efeitos não são muito comuns mas se fazem visíveis por exemplo nas cores particularmente brilhantes observáveis em certos escaravelhos.

A polarização acontece na reflexão do raio luminoso na qual embora o ângulo de reflexão coincida com o ângulo de incidência do raio luminoso, raramente a distribuição das ondas eletromagnéticas é a mesma pois alguns tipos de movimentos ondulatórios do fóton são suprimidos e o raio passa a ser polarizado. Para o homem na maioria dos casos este fenômeno passa imperceptível mas certas espécies animais baseiam sua orientação de vôo

3.2 A Textura e o Fenômeno de Iluminação

ou nado detetando o sentido de polarização dos raios solares. Em nosso meio são comuns os vidros polarizados que deixam preferencialmente passar certas orientações. A polarização pode ou não envolver um efeito brilhante acentuado como o “brilho molhado” do sol sobre estradas asfaltadas ou a polarização sem brilho que caracteriza a aparência do céu.

3.2.4 A Cor

As teorias que explicam a luz também são básicas para compreender uma outra característica óptica associada à textura: a Cor. Visto que a luz branca é composta por “raios luminosos de várias cores”, a presença ou ausência de alguns deles definem a cor do objeto. Em particular, existem dois mecanismos básicos de produção de cor: absorção e transmissão.

No fenômeno de cor por **absorção da luz**, os mecanismos de produção de cor são:

- por refração: A luz refrata-se e gera vários tipos de luz, tal como o fenômeno do arco-íris;
- por dupla reflexão: Interferências ópticas de natureza construtiva e destrutiva redirecionam a passagem de certos comprimentos de onda, gerando variação de cor;
- por Dispersão de Rayleigh: Ver 3.3.2;
- por remoção seletiva de certos comprimentos de onda. Para que isto aconteça a luz penetra o material e encontra espécies de absorvedores de energia. A luz refletida não possui mais os componentes absorvidos, dando assim sensação de cor; a cor do objeto é a cor complementar dos componentes absorvidos. Na Tabela 3.1 aparecem os comprimentos de onda absorvidos nas diferentes cores observadas.

A cor na maioria dos objetos (aqueles que não são fontes de luz mas meros refletores) obedece ao princípio de absorção (ver Figura 3.2), sendo este princípio muito usado em fotografia. A proporção relativa de absorção nas diferentes cores é mostrada na Tabela 3.2.

No fenômeno de cor por **transmissão da luz**, o mecanismo assemelha-se a usar três projetores com filtros vermelho-verde-azul sobre um material translúcido, podendo-se

Tabela 3.1: A Cor absorvida e observada para vários comprimentos de onda.

Cor observada	Cor complementar absorvida	Comprimento de onda absorvido, (nm)	Energia do fóton, (Kcal)
Nenhuma (a não ser fluorescente)	Ultravioleta	< 400	> 71.5
Amarelo esverdeado	Violeta	400-430	71.5-66
Amarelo alaranjado	Azul	430-490	66-59
Vermelho	Verde azulado p/ verde	490-510	59-66
Vermelho púrpura	Verde	510-530	56-53
Violeta	Amarelo esverdeado	530-560	53-51
Azul	Amarelo	560-590	51-49
Azul esverdeado	Laranja	590-610	49-47
Verde azulado p/ verde	Vermelho	610-700	47-42
Nenhuma	Infravermelho	> 700	< 42

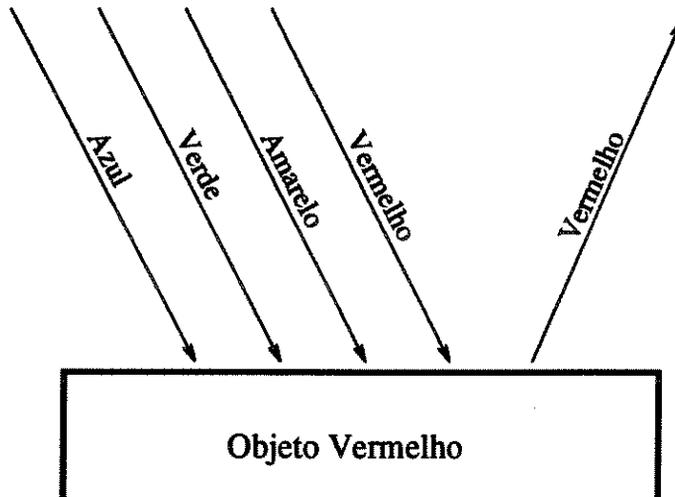


Figura 3.2: Um objeto vermelho pelo princípio de absorção.

Tabela 3.2: Absorção da Luz Visível para Várias Cores

Cor	Absorção relativa de energias visíveis
Preto	100
Azul escuro, verde-marrom	85-90
Cinza	75-85
Vermelho, marrons escuros, azuis claros	70-75
Creme	50-55
Branco	40-50

controlar a intensidade de cada um dos projetores. O processo de misturar aditivamente as cores gera a cor resultante. Tal mecanismo é usado nas telas dos dispositivos CRT (sigla em inglês para “tubo de raios catódicos”) e outros meios transparentes.

Mas a cor que percebemos no objeto não é exatamente a indicada nas experiências mencionadas. Além do comportamento em termos de refletância do material, ela depende da sensibilidade espectral do detetor de luz e da distribuição de energia da fonte luminosa, podendo-se formular que:

$$\text{Cor Real} = \text{Refletância Espectral do Objeto} + \\ \text{Sensibilidade Espectral do Olho} + \\ \text{Distribuição Espectral de Energia da Fonte Luminosa}$$

Isto significa que, mesmo que a cor¹ possa ser definida em termos de seu brilho (intensidade de luz), saturação (pureza relativa ou quantidade de luz branca misturada com o matiz) e matiz (atributo associado ao λ dominante na mistura de ondas da luz), a mesma cor não reluz igual quando aplicada em superfícies diferentes (por ex. metal, papel, tecidos, madeira, plástico ou vidro); e ainda, o mesmo objeto pode aparecer em cores diferentes quando observados ante fontes luminosas de distintas naturezas. Portanto, pode-se concluir que a aparência das superfícies está relacionada com as complexas formas de refletir a luz, com a quantidade de luz refletida e com a distribuição espectral de energia da fonte luminosa.

¹No modelo HSV

A cor da luz emitida pela fonte de luz pode também afetar a cor e, em geral, a aparência percebida nos objetos. Por exemplo os objetos têm aparências diferentes ao serem iluminados com luz natural visível e com luz ultravioleta, pois como mostrado na Tabela 3.3² acontecem distorções na cor natural dos objetos quando iluminados por fontes luminosas coloridas.

Tabela 3.3:O efeito da cor da fonte de luz na cor resultante dos objetos.

Cor natural dos objetos	Cor da Luz que ilumina o objeto					
	Red	Orange	Yellow	Green	Blue	Violet
black	red-black	orange-black	yellow-black	green-black	blue-black	violet-black
white	red	orange	yellow	green	blue	violet
gray	red shade	orange shade	yellow shade	green shade	blue shade	violet shade
red	red	scarlat	orange	brown	purplish black	reddish black
orange	red	orange	yellow-orange	greenish yellow	black	black
yellow	orange-red	yellow-orange	yellow	yellowish green	greenish black	black
light green	red shade	yellow-green	greenish yellow	green	blue-green	bluish shade
dark green	black	greenish black	yellowish green	green	greenish blue	blue-black
light blue	violet	dark gray	yellowish shade	blue-green	blue	violet
dark blue	purple	blue-gray	gray	blue-green	blue	blue-violet
violet	reddish black	red-purple	gray	blue	violet-blue	violet
purple	red shade	red shade	red shade	black	blue	violet
pink	red tint	red tint	red tint	greenish black	blue shade	violet shade

3.2.5 Fontes de Luz

Como já mencionado na seção anterior, a cor da fonte de luz influe na aparência percebida das superfícies. Para ver um objeto em sua cor plena, a fonte de luz deve produzir todos os comprimentos de onda da luz que são característicos dos *true-colors*³ do objeto, ajustados na mesma intensidade. Assumindo que a resposta do olho seja constante, na Figura 3.3 pode-se apreciar claramente as diferenças devido à fonte de luz.

²Os nomes das cores nesta tabela foram mantidos em inglês ([Bri80], página 70) devido ao fato de sua tradução gerar neologismos no português.

³Cor natural do objeto, resultado das características do material de que é formado ou recoberto.

3.2 A Textura e o Fenômeno de Iluminação

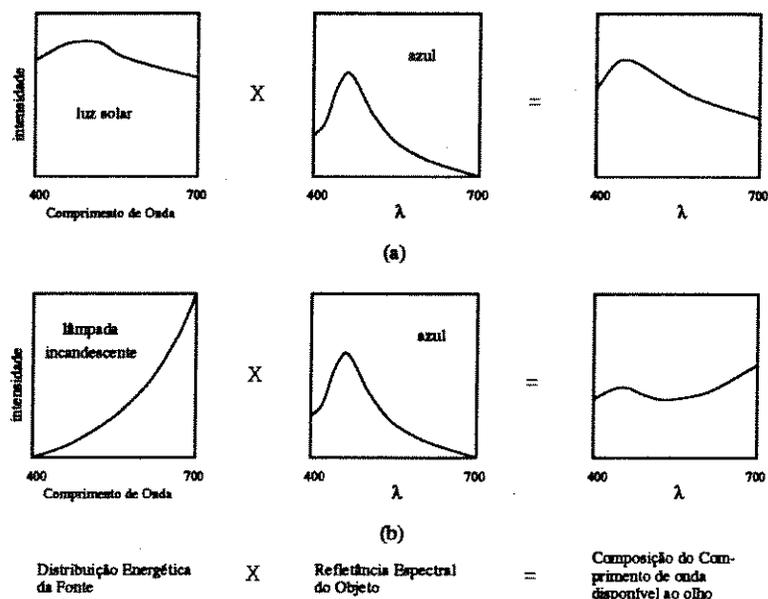


Figura 3.3: Onda luminosa resultante para fontes com diferentes distribuições espectrais de energia: (a) Luz solar; (b) Lâmpada incandescente.

Também o ângulo de projeção dos raios luminosos é um fator importante na aparência das superfícies. Quando a iluminação é normal, os raios incidem aproximadamente em ângulo reto, neste caso pequenas rugosidades da superfície podem passar imperceptíveis, pois quase não aparecem sombras. Num outro caso, quando a luz é projetada através da superfície posicionando-se a fonte de iluminação ao lado do objeto, percebe-se claramente as sombras e contornos devido à textura, isto é conhecido como **iluminação de rastreamento**.

Em geral, considerando-se a fonte de luz como uma fonte de radiação de energia luminosa na qual não acontece perda de energia por absorção ou transmissão, vê-se que, em termos do *rendering* da cor e dos efeitos nas propriedades do material, é de extrema importância observar: a natureza da fonte de iluminação (caracterizada pela curva de distribuição espectral de energia), a cor do raio luminoso (caracterizada pelo comprimento de onda do raio), a geometria da fonte de luz (se é um radiador pontual ou se possui uma área de radiação), a posição da fonte (para o cálculo do ângulo de projeção) e a intensidade ou eficácia dos raios luminosos.

Exemplos de fontes de luz são: a luz do sol, lâmpadas incandescentes, fluores-

centes, lâmpadas de sódio, de mercúrio ou a de um bulbo de flash. Estas fontes, a exceção do sol, precisam de energia elétrica para produzir luz. O sol tem uma excelente cor de *rendering*, pois possui uma distribuição espectral de energia relativamente uniforme, com todos os comprimentos de onda; já a lâmpada incandescente possui pouca eficiência e sua distribuição espectral de energia depende de sua potência em watts. As lâmpadas de mercúrio e sódio possuem muito maior eficiência, mas o *rendering* da cor é muito pobre devido à distribuição de energia descontínua na faixa visível.

3.3 A Textura e os Materiais

A velocidade e a direção dos raios luminosos, cujos valores são constantes no vácuo, podem sofrer alterações quando os raios encontram-se com os materiais. As características próprias de cada material determinam o comportamento do raio luminoso. Por isto, do ponto de vista microscópico “os átomos e moléculas dos materiais são sempre afetados de alguma forma por quase todos os tipos de radiação ” ([Bri80] pag.7) e ainda “os eventos registrados como resposta da matéria submetida a certas radiações também são mecanismos de produção desse tipo de energia”. Na faixa da luz visível, na maioria dos casos, o raio luminoso penetra a matéria causando excitação nos elétrons que se encontram nos níveis menos energéticos do átomo ou molécula. Quanto maior o comprimento de onda menos penetrantes são os raios. Os efeitos da radiação eletromagnética a nível de átomo e molécula são patentes na Tabela 3.4 (fonte [Bri80]).

Na abordagem microscópica estuda-se o percurso do raio luminoso dentro da matéria. Visto que ondas eletromagnéticas comportam-se diferentemente em materiais diferentes, a óptica (parte da física que estuda estes fenômenos) modela a interação na estrutura interna assinalando propriedades aos materiais segundo a relação entre o comportamento da onda no material e no espaço livre. Estas propriedades são : transmitância, refletância, índice de refração, homogeneidade óptica e isotropia as quais são funções que podem variar com o comprimento de onda do raio incidente.

Na aplicação destas propriedades são usadas as equações de Maxwell para medir a importância dos campos vetoriais . A partir delas os materiais são classificados como: dielétricos, cristais, metais (condutores), semi-condutores e magnéticos. Um dielétrico

Tabela 3.4: Eventos moleculares induzidos por radiações de diversos comprimentos de onda.

Comprimento de onda	Frequência, Hz	Nome da Região	Resposta dos átomos e moléculas
$(3 \times 10^8) - (3 \times 10^5)$ m	$1 - 10^3$	Forças Mecânicas	Nenhuma
$(3 \times 10^5) - 300$ m	$10^3 - 10^6$	Áudio	Nenhuma
300 - 3 m	$10^6 - 10^8$	Rádio-frequência	Translações moleculares, reorientações nucleares
$3 - 10^{-4}$ m	$10^8 - (3 \times 10^{12})$	Microonda	rotações moleculares, reorientações de elétrons
$10^5 - 700$ nm	$(3 \times 10^{12}) - (4 \times 10^{14})$	Infravermelho	Vibrações moleculares e efeitos de aquecimento direto
700 - 400 nm	$(4 \times 10^{14}) - (7 \times 10^{14})$	Visível	Transição eletrônica de baixa energia em nível de valência
400 - 10 nm	$(7 \times 10^{14}) - (3 \times 10^{16})$	Ultravioleta	Transição eletrônica de alta energia em nível de valência
10 - 0.03 nm	$(3 \times 10^{16}) - 10^{19}$	Raio-X	Transição eletrônica nos níveis internos; difrações pelos átomos
$0.03 - (3 \times 10^{-4})$ nm	$10^3 - 10^6$	Ultravioleta	Transições nucleares

caracteriza-se pelo seu índice de refração o qual mede como a velocidade do raio de comprimento de onda λ decresce no material em relação ao vácuo. Os metais ou condutores são descritos por seu índice de refração e seu coeficiente de extinção k (para um dielétrico k é por definição zero). Mesmo assim, na prática nenhum material pertence perfeitamente a apenas uma classe e mais ainda, as propriedades ópticas do material podem variar no espaço e segundo a direção com que o raio luminoso o atingiu. Considerando esta perspectiva, os materiais também são classificados como:

- **Homogêneos** - têm as mesmas propriedades ópticas em todos seus pontos.
- **Isotrópicos** - as propriedades ópticas do meio são independentes da direção da luz.
- **Anisotrópicos** - As propriedades ópticas são dependentes da direção da luz.
- **Normais** - É um meio homogêneo e isotrópico, no qual a luz se propaga uniformemente em todas as direções com velocidade constante.

Outras medidas importantes no estudo da aparência óptica dos materiais são:

- o índice de refração ;
- o tamanho das partículas de matéria;
- o comportamento da luz no material quanto aos fenômenos de reflexão-absorção-transmissão.

3.3.1 Índice de Refração

Para caracterizar o comportamento da variação da direção do raio de luz quando penetra um material usa-se o conceito de índice de refração, dado por:

$$\eta = C_{\text{vácuo}}/C_{\text{material}}$$

onde C é a velocidade do raio luminoso.

Este índice é característico de cada material e tem um papel importante na reflexão, refração, transmissão e polarização da luz pelo material.

Para entender a aparência dos objetos é importante entender que a quantidade de luz dispersada ou refletida numa interface composta por dois materiais depende fortemente da diferença entre os índices de refração dos meios. Por exemplo, as interfaces

ar-superfície e pigmento-veículo são tipicamente dispersadoras de luz. Se a diferença no índice de refração é grande no caso de uma interface ar-superfície, então a superfície será um bom dispersor e sua transparência é pequena. Se não existe diferença, então a luz não será dispersa e poderá ser totalmente absorvida ou transmitida a outros objetos depois de atravessar o material.

3.3.2 Efeitos do Tamanho da Partícula na Aparência do Material

Se o tamanho da partícula no material é menor do que um décimo do comprimento de onda da luz que o atinge, então a partícula não é capaz de dispersar a luz, o raio luminoso contornará a partícula e se dispersará de maneira diferente a como o faz com partículas maiores.

A teoria de dispersão da luz no caso de partículas pequenas foi desenvolvida por Lord Rayleigh em 1871 e é conhecida como a *Dispersão de Rayleigh*.

Rayleigh estabeleceu que para pequenas partículas a intensidade de luz dispersada em ângulos retos, com respeito à direção de incidência do raio luminoso varia:

1. diretamente com a intensidade da luz incidente;
2. diretamente com o quadrado do volume médio da partícula que dispersa;
3. inversamente à quarta potência do comprimento de onda da luz incidente.

A partir da última afirmação pode-se então dizer que comprimentos de ondas menores são mais dispersados. Por exemplo, raios de luz de 400nm são dispersados aproximadamente dez vezes mais efetivamente que raios luminosos de 700 nm. Assim também para explicar a cor do céu, a teoria de Rayleigh estabelece que o céu não é inerentemente azul, mas como pequenas partículas de água e pó de tamanho aproximado a 0.1λ (λ do espectro da luz visível) acham-se disseminadas na atmosfera, quando os raios do sol encontram estas partículas só as de menor comprimento de onda são dispersadas com eficiência em direção ao observador e, em consequência, o observador perceberá predominantemente os comprimentos de onda menores quando o céu é visto em ângulos retos. O fato de observar a cor azul e não violeta está relacionado com a menor sensibilidade dos nossos olhos ao violeta em relação ao azul.

Também o tamanho da partícula pode afetar a cor do material, pois como apenas a luz absorvida produz cor, materiais de partículas muito pequenas, em que a luz é dispersada muito eficientemente, podem perder sua cor (caso do cobalto azul, malaquita, etc). Por exemplo, na preparação de pinturas, o tamanho da partícula no pigmento e no veículo da pintura são considerações importantes para o seu preparo, pois afetam a rugosidade, brilho e uniformidade da pintura.

3.3.3 Caracterização dos Materiais a partir dos Fenômenos de Reflexão, Transmissão e Absorção

A estrutura dos materiais e os processos de transformação neles aplicados determinam aparências diferentes nas superfícies dos materiais. Empiricamente a caracterização destas diversas estruturas é feita através do estudo do percurso do raio luminoso quando incide sobre as mesmas. Distintas estruturas determinam comportamentos diferentes quanto à reflexão, transmissão e absorção dos raios luminosos, podendo-se assim definir “tipos de superfícies”.

Por exemplo, com relação à reflexão, uma superfície é dita *lisa* ou *especular*, quando os raios refletidos concentram-se numa direção (reflexão especular), tal como na Figura 3.4-a. Nesta situação a superfície se vê muito brilhante na direção (em relação à normal) análoga à de incidência. Quando acontecem pequenos desvios em relação a esta direção se diz que a superfície não é um perfeito espelho e que se trata de uma superfície lustrosa (do inglês *glossy surface*), estes casos já foram tratados em [TS67]. Quando se deseja caracterizar superfícies com este comportamento o grau de alisamento da superfície é usado como variável para controlar a luz refletida.

Uma superfície *semi-opaca* possui uma reflexão especular significativa, mas com suficiente reflexão difusa para atenuar qualquer brilho. Este tipo de superfície é notória em metais polidos. A Figura 3.4-b corresponde a este tipo de superfície a qual é característica de certos tipos de papel e de superfícies pintadas.

A superfície é chamada de *opaca*, *mate* ou *difusa* se a quantidade de luz refletida especularmente confunde-se em intensidade com a reflexão difusa. Nestas a aparência superficial tem pouca dependência com o ângulo de visualização já que a luz é uniformemente

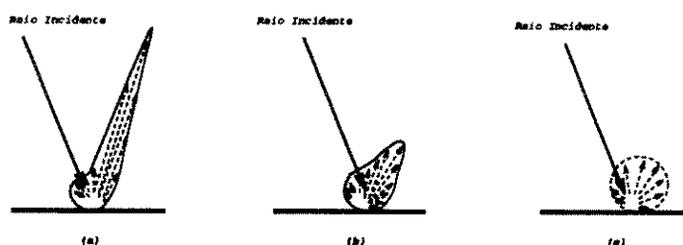


Figura 3.4:(a) Superfície lisa; (b) Superfície semi-opaca; (c) Superfície opaca.

dispersada em todas as direções. A Figura 3.4-c mostra uma superfície opaca própria de pinturas pastéis.

Com relação ao fenômeno de transmissão, se o raio luminoso que penetra a superfície é transmitido quase que totalmente e de forma concentrada, veremos uma superfície transparente. Quando os raios transmitidos se distribuem difusamente devido a uma certa rugosidade da superfície e o material possui as mesmas características de transmissividade por ambos lados da superfície, esta é chamada de translúcida.

Com relação à absorção, esta acontece quando existem moléculas absorventes na estrutura do material, como por exemplo moléculas de pigmento corante. Em materiais dielétricos nos quais as moléculas estejam isotropicamente distribuídas, a intensidade segue a Ley de Beer:

$$I(\lambda) = I_0(\lambda)e^{-C(\lambda)t}$$

Onde $I_0(\lambda)$ é a velocidade inicial, $I(\lambda)$ é a velocidade após percorrer uma distância t dentro do material e $C(\lambda)$ é uma constante positiva que descreve o comportamento das moléculas de pigmento corante. Também nos metais acontece absorção, neste caso sem a presença de corantes, mas nestes a absorção acontece tão rapidamente que a aparência é opaca a quase qualquer espessura. Existem casos em que o metal pode-se mostrar transparente e isto estaria determinado pela lei de Lambert:

$$I(\lambda) = I_0(\lambda)e^{-4\pi kt/\lambda}$$

onde k é o coeficiente de extinção do metal para um dado λ .

A caracterização destes vários tipos de superfície é bastante genérica e a proporção entre os raios concentrados e difusos assim como as frações de energia luminosa

refletida, transmitida e absorvida é própria de cada material . Para alguns materiais já foram empiricamente determinadas suas curvas espectrais ou funções que definem o comportamento de absorção, refletância e transmitância para os diversos comprimentos de onda da luz visível assim como os índices de refração dos mesmos o qual facilita a modelagem principalmente quando se trata de superfícies homogêneas.

Mas na maioria das texturas o comportamento dos raios luminosos é mais complicado devido à não homogeneidade dos materiais. Esta não homogeneidade pode ser visível só na superfície do material ou nas estruturas internas. Neste segundo caso, se o material tem a estrutura de camadas onde as camadas mais superficiais possuem algum grau de transparência e cada camada tem propriedades ópticas diferentes, as interfaces entre camadas produzem comportamentos particulares no transporte da energia luminosa. Este é o caso das superfícies sobre as quais foi colocada uma película transparente (caso de superfícies envernizadas). Quando a luz incide sobre elas o raio penetra a capa transparente e logo é difusamente refletido pelas partículas de pigmento da superfície interior, acontecendo uma absorção seletiva para projetar a impressão de cor. As características finais da superfície produzem-se pelas diferenças entre a luz refletida da capa mais exterior e a refletida na capa interior. A reflexão especular da luz branca é misturada com a reflexão difusa das capas subjacentes para dar a impressão de textura e cor. Também filmes transparentes podem criar efeitos de cor na forma de padrões de interferência como resultado da amplificação seletiva e supressão das intensidades de diversos comprimentos de onda da luz.

Outras propriedades são observáveis em materiais como os tecidos, em que a textura é influenciada pela orientação geométrica e densidade dos elementos componentes. Em geral, isto faz referência à forma e qualidade de fio usado, à geometria do entrelaçado e aos processos mecânicos aplicados. O brilho relaciona-se com o brilho de cada fio e das capas superficiais brilhantes aplicadas sobre o tecido; finalmente, a transparência está relacionada com a densidade do fio e com a transparência e densidade do entrelaçado. Uma textura tipo Moiré⁴ é alcançada criando-se pregas ou dobraduras de modo que as linhas de fio não fiquem paralelas. O tecido é logo acetinado para que o brilho de um fio crie impressões nos outros e ao desdobrar-se as linhas de fio, formem um padrão ondulado lustroso; as interferências e reflexos da luz sobre as linhas paralelas desalinhadas criam o efeito óptico.

⁴Tipo de tecido bastante conhecido por possuir um efeito lustroso bem particular.

Em casos como estes em que o comportamento das propriedades ópticas é bastante complexo, fica evidente que o comportamento dos raios luminosos precisa ser descrito por partes, por exemplo proceduralmente, de forma a incluir em cada procedimento detalhes do comportamento espacial de cada característica óptica, esta abordagem é posteriormente apresentada em 4.2.5.

3.4 A Textura e o Nosso Sistema de Visão

3.4.1 A Visão Humana - Aspectos Fisiológicos

Nem todos os animais têm a mesma visão. As abelhas enxergam radiações próximas à ultravioleta e com isto conseguem perceber coisas que o ser humano não vê. O ser humano consegue captar, em média, comprimentos de onda entre 400-700 nm e é capaz de separar os sinais da luz branca especularmente refletida daqueles sinais luminosos que dão idéia de cor: brilho, transparência, rugosidade, etc, de maneira a perceber profundidade e riqueza de detalhes na textura total.

Nosso sistema de visão caracteriza-se por possuir foto-receptores, um campo visual, acomodação das lentes do olho para perto-longe e adaptação para vários níveis de iluminação. Pode-se distinguir dois tipos de visão; a visão colorida, que acontece na retina (não na parte central do olho), através dos cones⁵ e a visão obscura ou visão estocópica, através dos cilindros⁶. Provavelmente a sensibilidade para variação de cor é causada por substâncias sensitivas com absorção seletiva, que transferem a energia luminosa às células receptoras. O mesmo acontece com a visão obscura (cilindros), onde se formam estas substâncias, porém sua composição e distribuição são desconhecidas.

Para a visão colorida existem no olho unidades moduladoras com sensibilidade mais acentuada para comprimentos de onda de 4000Å - 5550Å - 7800Å (os três campos espectrais R G B). Isto faz sentido do ponto de vista de produção de cor, pois, segundo Sölve Stenström (em [Ste64] pág 78), estabelece-se que “a impressão de todas as cores do espectro pode reproduzir-se, pelo menos aproximadamente, pela mistura de três cores: o vermelho ($\lambda = 4600\text{Å}$), o verde ($\lambda = 5300\text{Å}$) e o azul ($\lambda = 6500\text{Å}$)” e por outro lado, Young

⁵tipo de célula na retina dos olhos, assemelha-se a um cone e é especializada para a visão colorida

⁶tipo de célula de forma cilíndrica especializada para enxergar em ambientes obscuros

e Humholtz afirmam que “o sentido de cor em nosso sistema de visão é composto por três tipos de cores, cada um com sua própria região de sensibilidade e com sensibilidade máxima equivalente ao espectro das três cores básicas”.

Na visão obscura o olho possui maior sensibilidade para comprimentos de onda de 5070Å e 6500Å e a gama de cores que pode ser observada neste tipo de visão é limitada.

Existe também a chamada sensibilidade de contraste, que permite diferenciar entre campos adjacentes e ainda, segundo as experiências de Granit, parece existir um componente receptor que é sensível a todo o espectro visível de radiação e que dá a sensação de “branco” (este é chamado de “componente dominador”).

3.4.2 O Modelo simplificado de Câmara Fotográfica

Em síntese de imagens procura-se gerar num plano (*image-plane*) a projeção de um ambiente modelado tri-dimensionalmente (*object-space*), procedimento este bastante similar à aquisição de uma fotografia. O modelo mais simplificado de câmera fotográfica é o chamado de “pin-hole câmera”. Na verdade este modelo é uma simplificação do que acontece na visão humana onde os raios luminosos provenientes dos objetos estimulam as células de nossa retina (cones e cilindros) para a percepção da imagem através de uma abertura adaptável chamada de íris.

A “pin-hole câmera” é na verdade uma pequena caixa cujo interior está totalmente protegido dos raios luminosos e que no interior de uma de suas faces, uma lâmina de filme fotográfico está revestindo a superfície. Na face oposta à revestida pelo filme, é feito um pequeno furo pelo qual os raios luminosos têm passagem no momento da aquisição da fotografia, sendo que o fechamento imediato desta abertura é necessário para não prejudicar a qualidade da foto. O objetivo deste procedimento é captar somente aquela parte da cena que nos interessa, restringindo os raios luminosos refletidos de objetos fora do escopo de visão através de uma sincronização entre a cena que o observador vê e a passagem instantânea dos raios luminosos que entram pela abertura da câmera [UU80].

A simulação do nosso sistema de visão leva à modelagem de um sistema óptico que considera variáveis como a direção da luz, a distância à fonte luminosa, quantidade de energia luminosa que passa pelo sistema óptico, vértice da superfície de refração, raio de

curvatura, identificação das superfícies, etc. Se pensamos no conjunto câmera observador como o sistema de percepção da textura outras variáveis entrariam em consideração como tempo de aquisição da foto, diâmetro de abertura para o diafragma da câmera, o uso de filtros na lente, granulidade do fotoreceptor, etc. Em relação ao observador considera-se a posição deste em relação ao objeto, as dimensões da janela de visualização, etc.

3.4.3 A Abordagem Psicologista de Percepção da Textura

Fonte de grande interesse para este estudo tem sido as pesquisas ao redor da percepção humana, em particular os psicologistas [BJ83] [Jul75] [Tre85] [Not85] [Ram82] quem lidam com o fenômeno de percepção da textura.

Nesta abordagem a textura é vista como uma tarefa que o sistema visual humano realiza a qual “envolve puramente processos de percepção”, isto é, “tarefas que podem ser realizadas espontaneamente e não precisam da realização de processamentos cognitivos que impliquem escrutínio” ([Jul75]). Com efeito, a percepção da textura é alcançada nas etapas iniciais do processo de visualização a saber nos primeiros centos de milissegundos de inspeção ; tempo impossível de atingir se perceber textura envolvesse um escrutínio linha por linha da imagem, considerando-se principalmente que são necessários aproximadamente 200 milissegundos para gerar movimento ocular voluntário.

Assim, desde a primeira impressão visual é possível extrair características como cor, brilho, rugosidade, direcionalidade, homogeneidade, forma das primitivas etc. identificando seu comportamento no espaço da superfície observada. O número das características apreciáveis é limitado e ainda não totalmente investigado, o que se sabe é que “elas são mapeadas sobre áreas diferentes do cérebro onde cada área é especializada em analisar uma propriedade diferente”.([Tre85])

A extração das características visuais corresponde à decomposição da imagem e portanto: “a percepção destas características pode ser representada isoladamente e assim descrita no espaço de resposta para simular a textura total”.

Com base nisto Englert e Sakas [ES89] definem textura visual como: *a impressão sensorial obtida pelos olhos, tratando-se portanto de uma impressão visual, de forma tal que duas áreas vizinhas dentro de um campo visual podem ser espontaneamente identificadas*

através de uma observação com olhos fixos.

Destas considerações ve-se que: o ser humano identifica separadamente sinais de brilho, cor, transparência, rugosidade, etc. (capacidade para distinguir características ópticas simultâneas); que o olho humano possui sensibilidade de contraste para diferenciar campos adjacentes diferentes (capacidade para caracterizar o comportamento do sinal variando no espaço) e que a variedade de cores que o ser humano é capaz de reconhecer é derivada de frequências centrais ao redor de valores semelhantes as frequências básicas do sistema RGB e que a simulação da textura total pode ser feita a partir da modelagem separada de cada sinal, sendo que na apresentação simultânea dos mesmos o observador é capaz de integrá-los na textura do objeto.

3.5 *Rendering* e o Modelo de Iluminação em Síntese de Imagens

Todos os conceitos revisados nas seções anteriores são associados para determinar a radiância de um ponto P em uma determinada superfície através do que se conhece como o **Processo de Rendering**. O princípio básico da conservação da energia usado na transmissão de calor foi adaptado por Kajiyama [Kaj86] ao fenômeno de iluminação formulando-se assim o que é conhecido na literatura como a *Equação de Rendering*:

$$L_i(P, \psi_s) = L_e(P, \psi_s) + L_r(P, \psi_s)$$

com

$$L_r(P, \psi_s) = \int_{\Omega'_P} \rho(P, \psi_i, \psi_s) L_i(P', \psi_i) \cos\theta_i dw$$

onde: $L_i(P, \psi_s)$ é a radiância que o observador pode enxergar no ponto P da superfície tendo ψ_s como direção de observação, $L_e(P, \psi_s)$ é a radiância própria do ponto P e $L_r(P, \psi_s)$ é a radiância no ponto P proveniente dos raios refletidos desde outros pontos ao redor e que é observável desde a direção ψ_s , sendo que: $L_i(P', \psi_s)$ é a radiância incidente sobre todos os pontos P' com vista ao hemisfério superior a P e $\rho(P, \psi_i, \psi_s)$ representa a refletância da superfície dada uma direção de incidência e uma direção de observação.

Em computação gráfica este processo modela a interação dos fatores Luz-material-sistema_de_visão e neste contexto, um **Modelo de Iluminação** é um método que resolve a

equação acima considerando certas condições. Os modelos até agora desenvolvidos (p. e. [Pho75] [Bli77] [Bli82] [TS67] etc.) não conseguem representar toda a complexidade do fenômeno de iluminação e mesmo as soluções mais detalhadas enfrentam um compromisso entre ganho em realismo e custos de processamento. Em geral estes modelos procuram representar o fenômeno de iluminação por modelação indireta, isto é, não se pretende simular os elementos intrínsecos que geram um dado comportamento do raio luminoso e sim, introduzir num modelo simplificado (geralmente os que se servem da óptica geométrica), parâmetros com os quais seja possível manipular a direção e distribuição dos raios luminosos incidentes, refletidos, transmitidos e absorvidos de formas concentrada e difusa. Estes parâmetros são também conhecidos como **parâmetros de tonalização**. A função de textura, cujos elementos variáveis já foram discutidos na seção anterior, neste contexto buscará introduzir elementos que façam possível esta manipulação visando a representação da textura total. Neste trabalho o estudo será limitado aos modelos atualmente implementados no ProSim que resolvem a equação de *rendering* pelo método de ray-tracing, a saber uma versão modificada do modelo de Whitted-Phong [Whi80][Mad91] e do modelo de Torrance-Sparrow [TS67].

A versão modificada do modelo de iluminação de Whitted-Phong tem base empírica e responde à seguinte formulação geral:

$$I = I_p k_d O_d + S f_{att} I_p [k_d O_d (\vec{N} \cdot \vec{L}_i) + k_s (\vec{N} \cdot \vec{H})^n] + k_r I_r + k_t I_t \quad (3.1)$$

onde:

I_p é a luminosidade própria do ponto P ,

I_r é a intensidade do raio refletido.

I_t é a intensidade do raio transmitido e refratado,

k_d é o coeficiente de difusão da superfície,

k_s é o coeficiente de especularidade da superfície,

k_t é o coeficiente de transmissão da superfície,

n é o expoente especular da lustrosidade,

S é uma função delta para introduzir o efeito de sombreamento,

f_{att} é o fator de atenuação devido à distância à fonte de luz,

O_d é uma função que caracteriza o comportamento da superfície para a componente difusa de refletividade,

\vec{N} é o vetor unitário normal à superfície no ponto P ,

\vec{L} é o vetor unitário na direção de incidência do raio luminoso desde a fonte de luz,

\vec{H} é o vetor bissetor entre \vec{L} e o vetor unitário na direção de observação.

O modelo original de Phong desenvolve soluções para superfície perfeitamente especulares ou perfeitamente difusas mas não considera casos intermediários. Também não inclui tratamento para superfícies transparentes. O modelo acima é uma versão modificada em que o parâmetro de transparência é incluído mas que não considera reflexões difusas na transparência (efeito de translucidez).

O modelo de Torrance-Sparrow [TS67] é um modelo de base física que em 1981 foi adaptado para computação gráfica por Cook e Torrance [CT81]. Os modelos de base física manipulam uma base conceitual que usa termos como energia radiante, intensidade radiante, radiância, irradiância, etc. cujas definições podem ser achadas em [Bri80] ou em [Bes92]. Estas magnitudes são susceptíveis de ser medidas fisicamente e resultados experimentais encontram-se registrados na literatura pertinente (p.e [Pal85]) o que permite que os dados disponíveis possam ser introduzidos no modelo na tentativa de representação do fenômeno de iluminação com os objetos do mundo real. A equação geral do modelo de iluminação de Torrance-Sparrow tem a forma:

$$I = \rho_a I_a + I_i (\vec{N} \cdot \vec{L}) dw_i (k_d \rho_d + k_s \rho_s) \quad (3.2)$$

Neste modelo para introduzir as variáveis associadas ao material que influenciam o comportamento dos raios luminosos, o termo correspondente à componente especular de refletividade é melhor expresso como:

$$\rho_s = \frac{F}{\pi} \frac{DG}{(\vec{N} \cdot \vec{L})(\vec{N} \cdot \vec{L})},$$

onde:

D é uma função que representa a distribuição das microfacetas na superfície,

G é uma função que modela a atenuação geométrica dos raios refletidos devido à interferência causada pela presença das microfacetas na superfície,

F é o termo de Fresnel.

Um bom resumo das soluções propostas para a implementação destas funções encontra-se em [FvFH92].

Como a técnica de *ray-tracing* [Gla88] supõe que a luz pode ser simulada como raios de linhas retas lançados em todas as direções, os modelos geralmente associados a esta técnica não são capazes de representar fenômenos associados à natureza ondulatória da luz como os tratados em 3.2.3, não sendo os modelos acima citados uma exceção.

3.6 Resumo

Neste capítulo estudamos o fenômeno físico por detrás da textura pretendendo identificar os modelos que a física e em particular a óptica, fornecem para a representação do comportamento dos raios luminosos na sua interação objetos-observador. Nestes modelos pode-se reconhecer boa parte das variáveis que se entrelaçam para produzir o efeito textura, algumas das quais são descritas na Seção 3.6.1.

Quandos raios luminosos são modelados como linhas retas com uma única cor resultante associada dizemos que nos movemos no campo da “óptica geométrica”, e esta abordagem é a usada na maioria da simulações por computador. Quando os raios são modelados espectralmente, como ondas senoidais que possuem faixas de frequências bem definidas (em termos de percepção visual a frequência está associada à cor) é possível modelar melhor efeitos com a difração, polarização e interferência, bem como dar um tratamento espectral à cor, ou seja, definir cor como a resultante das ondas luminosas, que atinge a retina do observador após refletir uma superfície.

Neste estudo se fez notório que a estrutura interna do material, o tamanho das partículas, bem como interfaces entre materiais (entenda-se como material meios sólidos, líquidos e gasosos) influenciam o comportamento dos raios luminosos incidentes e refletidos (*Bidirectional Reflectance Distribution Function BRDF*) e que a estrutura dos materiais não está sendo estritamente modelada mas somente simulada através da atribuição de constantes medidas fisicamente para caracterizar propriedades ópticas de cada tipo de material.

Um estudo da percepção visual humana nos levou a identificar qual é o mecanismo de percepção da textura. A abordagem psicologista propõe que ao ser humano lhe é natural perceber e descrever textura a partir da identificação e descrição das “características ópticas” reconhecidas. Vimos também que em termos de modelamento a percepção visual é simulada pelo modelo de câmera fotográfica e que este modelo possui variáveis que

caracterizam a imagem final percebida na projeção 2D de uma cena 3D.

Também, foi visto que o processo de *rendering* é uma solução ao problema de transporte de energia luminosa e que, associado ao processo de *rendering* existe um modelo de iluminação, através do qual as variáveis medidas fisicamente podem ser embutidas na produção de imagens sintéticas. Ficou evidente que cada algoritmo proporciona soluções através de uma simplificação do problema e desde este ponto de vista, o *ray-tracing* por exemplo, é uma técnica que desenvolve soluções para os pontos da cena contidos no *image-plane* assumindo que a energia luminosa se concentra sobre raios que em linha reta percorrem o caminho desde o observador até a fonte luminosa (*backward tracing*). O modelo de iluminação considerado pode ser simples ou bastante complexo. Por exemplo, no ProSim tem-se o modelo modificado de Whitted-Phong, que é um modelo de base empírica útil para representar certos tipos de superfícies e o modelo de Torrance-Sparrow como um modelo mais detalhado, que possui base física e com o qual se podem alcançar resultados mais precisos. Isto evidencia que nem todos os modelos são capazes de representar todos os efeitos e que isto tem repercussão na textura.

3.6.1 Variáveis úteis para criar efeitos na Textura

A partir das seções anteriores fica evidente a complexidade do fenômeno *textura* e que a caracterização do mesmo está relacionada a modelos físicos ou empíricos já existentes que refletem as relações entre luz, observador e material. Os relação a seguir procura mostrar que existem variáveis que estão sendo ignoradas na texturização de objetos sintéticos as quais podem ser muito úteis para a simulação de certos efeitos (embora, em termos de “modelagem”, uma certa simplificação seja sempre necessária).

Quanto à modelagem dos raios luminosos:

1. velocidade ou comprimento de onda do raio luminoso;
2. $\rho(\lambda) = \text{Energia luminosa refletida,}$
 $\tau(\lambda) = \text{Energia Luminosa transmitida,}$
 $\alpha(\lambda) = \text{Energia Luminosa absorvida,}$
ou gráficos de refletância espectral, transmitância espectral e absorção espectral;

3.6 Variáveis úteis na textura

3. modelo que implemente interferências ópticas;
4. distribuição espectral de energia da fonte luminosa;
5. cor da fonte de luz;
6. ângulo de projeção dos raios luminosos sobre a superfície (posição da fonte de luz, normal à superfície);
7. geometria da fonte de luz;
8. intensidade ou eficácia dos raios luminosos.

Quanto ao modelo do material

1. densidade do material;
2. transmitância;
3. refletância;
4. índice de refração;
5. homogeneidade óptica;
6. isotropia;
7. tamanho da partícula;
8. orientação geométrica e densidade dos elementos componentes.

Quanto ao modelo câmera-observador

1. sensibilidade espectral do detetor;
2. distância à fonte de luz;
3. quantidade de energia luminosa que passa pelo sistema óptico (abertura do diafragma);
4. tempo de exposição ;
5. granulidade do fotoreceptor;
6. posição do observador;
7. existência de filtros.

3.6.2 Definições para a Síntese de Texturas

A partir dos tópicos tratados neste capítulo, algumas definições e conceitos já podem ser estabelecidos a fim de delimitar o escopo deste trabalho e estabelecer a base conceitual do mesmo:

O conceito de textura na linguagem natural: Na linguagem comum, o conceito de textura está relacionado com as características apreciáveis na superfície dos objetos, que nos fornecem informação simultânea da variação local da rugosidade, brilho, cor, etc., no espaço da superfície. Na prática, o ser humano obtém informações de textura pelo sentido da visão e do tato e relaciona estas informações à natureza do material com que o objeto foi construído ou recoberto.

O conceito de textura restrito à visão: Por tratar-se neste trabalho de um ambiente de síntese de imagens, não interessará nesta abordagem simular os aspectos da textura que são percebidos pelo tato e sim os percebidos pela visão.

O conceito de Textura Observada: Aparece então o conceito de textura observada como a impressão visual, dependente da natureza das superfícies, das condições de visualização e das propriedades do sistema de percepção visual. Nesta textura observada é natural ao ser humano integrar as características ópticas que está percebendo: brilho, cor, uniformidade, transparência, rugosidade, etc. com suas variações no espaço da superfície do objeto ou região, numa única entidade que chama de textura. Também dentro deste conceito de textura observada é válido considerar textura a "impressão visual" sobre um conjunto de objetos. Este conceito fica mais evidente quando são tratadas áreas amplas de observação, como a textura observada sobre regiões pedregosas ou campos de cultivos numa imagem aérea. Perceba-se que o termo "impressão visual" existe desde que os estímulos vindos de uma superfície sejam captados pelos nossos órgãos de visão, processados no cérebro e finalmente integrados numa entidade que chamamos de textura, mas que na realidade identifica a textura observada.

O Campo de Textura: Como definido em 2.10.1.

O conceito de Característica Ótica: A textura observada integra aspectos visuais que chamamos de brilho, cor, uniformidade, transparência, rugosidade etc. e su-

as variações no espaço da superfície do objeto ou região que está sendo observada. Estes aspectos serão chamados de “características ópticas” e como já foi discutido aparecem como resultado da interação luz-material-sistema de visão . Em relação a este conceito, em toda textura sempre é possível reconhecer pelo menos uma característica óptica e nestes casos, essa única característica será considerada a textura do objeto.

O conceito de Resolução da Textura: Em alguns casos, pode-se reconhecer texturas diferentes quando o observador posiciona-se a distâncias diferentes do objeto ou área observada. Neste caso falar-se-á de resolução da textura.

O relacionamento Espaço-Objeto - Espaço-Textura: Neste trabalho não será feita uma pesquisa extensiva sobre o relacionamento entre estes espaços. Por enquanto este relacionamento será modelado unicamente por funções de mapeamento inverso de projeção linear, desenvolvidas para as diferentes primitivas de modelagem no espaço objeto.

O Domínio da Textura Observada: Embora a textura observada possa ser o produto da interação Energia-Luminosa-Material-Sistema de Visão, isto não significa que tudo seja textura. É fácil perceber que quando a *dimensão* das variações texturais extrapola certa faixa, essa variação deixa de ser chamada textura. A pergunta então surge naturalmente: a partir de onde o problema deixa de ser textura e passa a ser *modelamento*? ou até onde vale a pena mostrar os detalhes da textura se o efeito pode não ser observável? (por exemplo, o observador se posiciona a uma distância muito grande). A resposta a estas perguntas possui um caráter relativo e não é dada em dimensões do mundo real, mas em dimensões da imagem projetada da cena do mundo real (*image-plane*) com relação ao observador. Tem-se achado na literatura [WAT92] uma escala que pretende definir a questão e para este trabalho a consideraremos como uma boa referência ao decidir texturizar ou não (ver Figura 3.5).

Simulação da Textura: A simulação da “textura observada” trata-se então da criação de imagens na tela de dispositivo de visualização de forma a emitir estímulos análogos aos que excitariam o nosso cérebro, ao observar uma foto da cena que está sendo simulada. Assim a textura é simulada reproduzindo as características ópticas percebidas que, em última instância, no dispositivo de visualização, aparecerão como

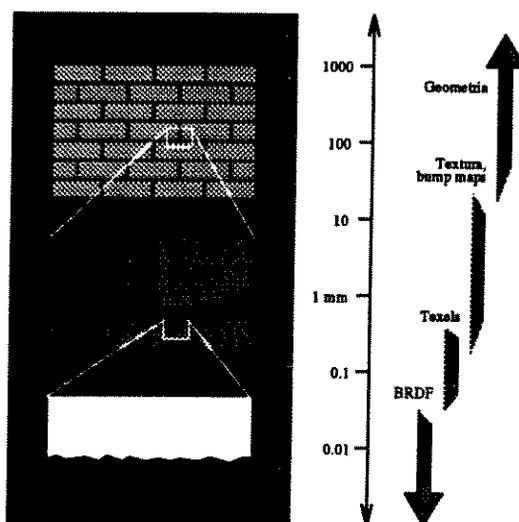


Figura 3.5: Escala de Texturização.

intensidades de cor sobre a superfície dos objetos sintéticos. Portanto, o problema básico de simulação consiste em determinar adequadamente a intensidade de energia luminosa a ser colocada em cada pixel do dispositivo de visualização. Isto é possível, pois os algoritmos de *rendering* incluem um modelo câmera-observador e um modelo de iluminação. Então, simular a natureza da superfície, entendida como o conjunto de propriedades geométricas e materiais que contribuem à aparência de textura, reduz-se a balancear apropriadamente os parâmetros que representam as interações de cada material com a energia luminosa - **parâmetros de tonalização**. A simulação das condições de visualização está diretamente relacionada com a quantidade e a forma com que a energia luminosa chega ao objeto e que um observador imaginário consegue enxergá-la. Quantidade e forma de radiação dos raios luminosos de chegada são modeladas considerando-se, por exemplo, o número de fontes de luz, a cor das mesmas, a sua posição, geometria, sua distribuição espectral de energia, etc. Quantidade e forma dos raios luminosos de saída, que um determinado observador consegue enxergar, estão relacionados com as características do material, a posição do observador, a janela de visualização, posição da fonte de luz, etc..

Capítulo 4

Uma Proposta para a Síntese de Texturas no ProSim

4.1 Introdução

Este capítulo apresenta uma proposta para a Síntese de Texturas no ProSim o que envolve: a definição de um Modelo Geral de Texturas (o qual pretende ser uma abordagem geral para o problema de sintetizar texturas), a especificação de uma estrutura para um Sistema de Texturas no ProSim, o processo que resolve o problema de sintetizar textura neste contexto e finalmente um exemplo com os resultados alcançados segundo esta abordagem.

4.2 Um Modelo Geral para Modelagem e Aplicação de Texturas em Ambientes Realísticos

Ter uma plataforma para o tratamento das várias técnicas de modelagem e aplicação de texturas revelou a necessidade de uma conceitualização geral a ser usada. Encontrou-se em Englert e Sakas [ES89] uma formulação adequada de um modelo geral para a síntese de texturas ao qual foram acrescentados novos conceitos e algumas modificações, produto da experiência da implementação de texturas no ProSim.

O modelo é dito geral porque permite a criação de texturas independentemente da técnica de modelagem dos objetos, do algoritmo de *rendering* e ainda possibilita a composição de mais de um modelo de textura, na produção de uma única textura final. Este modelo supõe: um comportamento do campo de textura que é independente do objeto que há de possuí-la, comportamento este definido num espaço apropriado (Espaço da Textura); funções f ; que descrevem o comportamento espacial das características ópticas da textura (Funções Parciais de Textura - as que quando possuem parâmetros de instanciação são chamadas de primitivas de textura); a possibilidade de compor uma nova textura a partir de outras (operações entre funções de textura); a integração das várias características ópticas numa textura final (Função Completa de Textura) e funções que relacionam de forma particular a textura ao objeto (Funções de Aplicação da Textura). A base conceitual considerada em nossa abordagem tomará os conceitos desenvolvidos em 3.6.2 aos quais serão acrescentados os descritos a seguir.

4.2.1 A Síntese de Texturas

A base para a síntese de texturas nesta proposta é o conceito de Textura Observada aliado à abordagem psicologista descrito em 3.4.3. Isto é, ao falarmos em síntese de texturas o que esta-se na verdade simulando é a textura tal como observada, isto envolve descrever e modelar o campo de textura através da modelagem e representação gráfica das características ópticas reconhecidas. De acordo com a abordagem psicologista, a capacidade “natural” do ser humano para reconhecer aspectos visuais e integrá-los numa entidade que é chamada de “textura observada”, leva-nos à facilidade de descrever a textura por partes e modelá-la através do comportamento das características ópticas. Isto envolve funções mais simples de determinar do que as necessárias para modelar a textura final diretamente. Esta proposta baseia-se em introduzir estas funções mais simples como elementos que modificando os parâmetros do modelo de iluminação e outros parâmetros do *rendering* (parâmetros de tonalização), consigam simular uma certa característica óptica. Desta forma, na imagem final a intensidade de cada pixel sera o resultado que o algoritmo de *rendering* atribui à presença simultânea das características ópticas presentes no ponto tratado. Um esquema geral desta idéia é apresentado na figura 4.1.

Deve-se observar que não necessariamente as diferentes características visuais

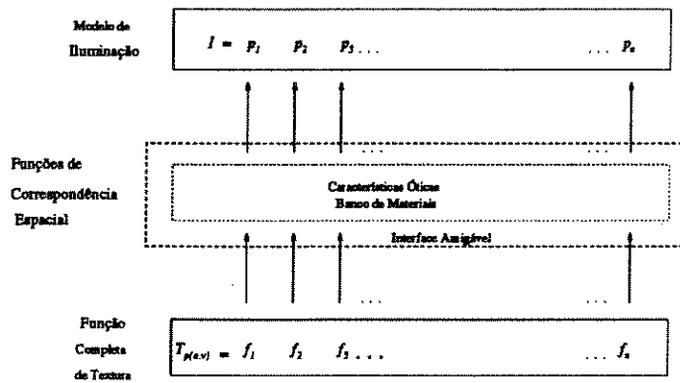


Figura 4.1: Interação das funções texturais no Modelo de Iluminação

que o ser humano distingue terão, no modelo de iluminação, um número equivalente de parâmetros que as representem. Os parâmetros considerados podem ser em maior ou menor número do que as características ópticas, dependendo de “o quê” eles signifiquem no contexto do modelo de iluminação.

4.2.2 Espaço da Textura

Em síntese de imagens os objetos são definidos num espaço tridimensional chamado de Espaço Objeto (de variáveis X, Y, Z). Mas a imagem que nós vemos na tela é uma projeção deste espaço num plano perpendicular à direção de observação, este plano é chamado de Espaço da Imagem e é geralmente expresso em variáveis (x, y) . Por outro lado a abordagem mais tradicional de definição de texturas [Cat75] trata da prototipação do campo de textura num espaço que é relacionado ao espaço do objeto através de funções de mapeamento. Este espaço conhecido como espaço da textura é normalmente definido pelas variáveis normalizadas (u, v) no caso bi-dimensional e (u, v, w) para o caso tri-dimensional.

Neste modelo redefine-se o conceito de espaço de textura, o qual, por enquanto, pode ser entendido como o sistema de coordenadas para a definição de funções de textura. As dimensões deste espaço não são pré-estabelecidas e a textura pode ser tratada como uma característica de superfície, função de um domínio bi-dimensional, pode ser estendida ao espaço 3D, como no caso da textura sólida [Pea85] [Per85], ou ao espaço 4D, se supusermos por exemplo, funções texturais para o movimento de um corpo no espaço e no tempo.

Definição: Se considerarmos n como o número de parâmetros de tonalização, usados para simular uma impressão de textura e sendo que no enfoque psicologista [BJ83], a percepção visual humana cria espaços de resposta para cada aspecto visual (cor, rugosidade, uniformidade, etc), então é possível definir sistemas de coordenadas para cada aspecto ou característica óptica (ch_i) de forma que o espaço de textura T pode ser definido como um espaço de coordenadas cartesianas tal que:

$$T = \mathcal{R}^m \times \prod_{i=1}^n ch_i$$

o qual é chamado de Espaço de Textura T . Onde \mathcal{R}^m representa o domínio m -dimensional de definição da textura, ch_i é o conjunto possível de valores da i -ésima característica óptica e $\prod_{i=1}^n ch_i$ o produto cartesiano dos domínios de todos os ch_i 's.

4.2.3 A Função Parcial de Textura

É chamada de função parcial porque ela não representa a textura final de objeto algum e sim o comportamento espacial de um ou mais dos parâmetros de tonalização, pretendendo assim simular o comportamento espacial de uma certa característica óptica notória na textura. Esta definida como:

$$f_i : \mathcal{R}^m \rightarrow ch_i$$

e é chamada de Função Parcial de Textura¹.

A função parcial de textura não está associada a nenhuma característica óptica em particular, sendo simplesmente um modelo de comportamento para o espaço da textura e pode ser usada para simular qualquer variação que nele ocorra.

A maioria de funções de textura f_i , podem ser implementadas com parâmetros (passando a ser chamadas de primitivas de textura) a fim de que com uma mesma primitiva possam ser produzidas instâncias diferentes da mesma função.

¹Note-se que, para o sistema de cores no ProSim (RGB), f_i na realidade será um vetor de funções tri-dimensionais que atribui valores às componentes de intensidade R-G-B de cada pixel.

4.2.4 O Mapa de Textura

É a aparência gráfica da função f_i que modela a característica óptica ch_i . O mapa de textura é a representação gráfica do comportamento espacial de uma certa característica óptica e pode ser gerado pela instanciação de alguma primitiva de textura. Pode ser denotado:

$$M_i = \{(a, b, c) : a, b \in \mathcal{R}, c = f_i(a, b)\}$$

onde f_i é uma função parcial.

O mapa parcial de textura pode ser simples (por exemplo quando pela execução de uma primitiva gera-se diretamente o mapa desejado) ou composto como quando se trata da composição de várias f_i 's, organizadas: hierarquicamente em níveis (ver 4.2.7), espacialmente numa matriz de mapas de texturas ou compostas através de algum procedimento.

4.2.5 A Função Completa de Textura

Como foi definido que a textura observada é a “impressão visual” resultante da percepção simultânea de um conjunto de características ópticas, então a textura final de um objeto consiste num conjunto de funções parciais de textura (ou um conjunto de mapas de textura) que juntas tentam simular o conjunto de características ópticas que ocorrem simultaneamente na textura final.

Assim a simulação total de uma textura, considerando-se o efeito final na imagem, está definida como a Função Completa de Textura ctf , onde:

$$ctf : \mathcal{R}^m \rightarrow T$$

que é expressado como a n -upla:

$$ctf = (p_1, p_2, f_1, f_2, \dots, f_n)$$

onde $(p_i$ para \mathcal{R}^2 , $i = 1, 2$) é a função projeção na i -ésima coordenada e f_i para os $i = 1, \dots, n$, é a função parcial de textura dos n parâmetros de tonalização.

Por exemplo, no espaço bi-dimensional, a função completa de textura do ponto (a, b) é descrita pela n -upla:

$$ctf(a, b) = (p(a, b), p(a, b), f_1(a, b), \dots, f_n(a, b))$$

Deve-se notar, com relação às ctf 's, que:

- podem ser estabelecidas composições entre várias funções de textura f_i denotando-se:

$$f_k(f_l) \text{ onde } k, l \in [1, n] \text{ e } k \neq l;$$

- para aquelas ctf 's diretamente mapeáveis sobre os parâmetros de iluminação, deve valer,

$$dom(ctf) = dom(f_i), \quad \text{com } i = 1, \dots, n$$

onde dom é o domínio da função.

4.2.6 Funções de Aplicação de Textura

Englert e Sakas [ES89], definem as funções de textura f_i como aquelas que afetam cada parâmetro de tonalização p_i , mas não fazem menção ao respeito da forma como estas funções afetam estes parâmetros. O valor da função textural avaliado em cada ponto do espaço pode estar afetando os parâmetros de cor, de transparência, ambos ou nenhum deles. Ainda mais, o valor obtido da função de textura pode estar sendo usado para substituir o parâmetro original, pode ser usado como um coeficiente para o parâmetro $(f_i \times p_i)$, pode ser somado ao valor corrente do parâmetro $(f_i + p_i)$ ou pode assumir qualquer outro relacionamento matemático $(Af_i + Bp_i + C)$, por exemplo).

Tendo isto em consideração no modelo aqui proposto observa-se que: uma vez estabelecido o valor numérico que provém da função parcial (f_i) para o ponto $P(x, y, z)$ em tratamento e os parâmetros p_i de tonalização a serem afetados, existem diversas maneiras de relacioná-los. Então, a forma como o valor que provém da função parcial de textura afeta o parâmetro de iluminação é chamada de Função de Aplicação da Textura e está definida como:

$$g_i : a \rightarrow b$$

onde:

a , é um número real, calculado pela função de textura f_i e pertencente ao intervalo $[0,1]^2$ caso a textura tenha sido modelada no espaço unitário;

b , é também um número real, sendo que seu intervalo de variação depende dos valores permitidos ao parâmetro p_i dentro do modelo de iluminação.

A função completa de textura estaria, então, sendo definida como :

$$ctf : \mathcal{R}^m \rightarrow T$$

que é expressado como a n -upla:

$$ctf = (p_1, p_2, g_1(f_1), g_2(f_2), \dots, g_n(f_n))$$

assumindo que cada ch_i tem uma representação associada dentro do modelo de iluminação (ou seja, parâmetros p_i que a modelem).

Na realidade, a função de aplicação $g_i(f_i)$ poderia ser considerada uma função de textura como um todo, mas o fato de considerá-la separadamente torna possível a introdução de variações valiosas para o ajuste da textura final, descompromissadas do processo de determinação do campo de textura base.

4.2.7 Relacionamento entre Funções Texturais

Uma textura pode ser descrita por vários níveis hierárquicos, onde o efeito final, é o resultado de operações sobre outras funções de textura.

Depois de gerada, uma textura pode ser transformada (por exemplo, através de filtros), ou combinada com outras texturas (como no caso de superposição de texturas), até

²Esta restrição é feita para padronizar o valor dos argumentos das g_i 's

a obtenção da textura final. Na Figura 4.2 cada bloco representa uma função completa de textura e as flechas representam as operações sobre as texturas.

Genericamente os tipos de operações possíveis entre funções f_i podem ser agrupadas como:

- G** Funções de Geração: para criar mapas de textura. Uma operação do tipo **G** consiste na verdade em executar a primitiva que gera um mapa parcial de textura para uma certa característica óptica ou definir os parâmetros de geração da textura final (**T**).
- T** Funções de Transformação: modificam uma função, por exemplo, através de filtros;
- C** Composições: definem operações sobre funções parciais de textura ou procedimentos que combinem de forma particular mapas de textura.

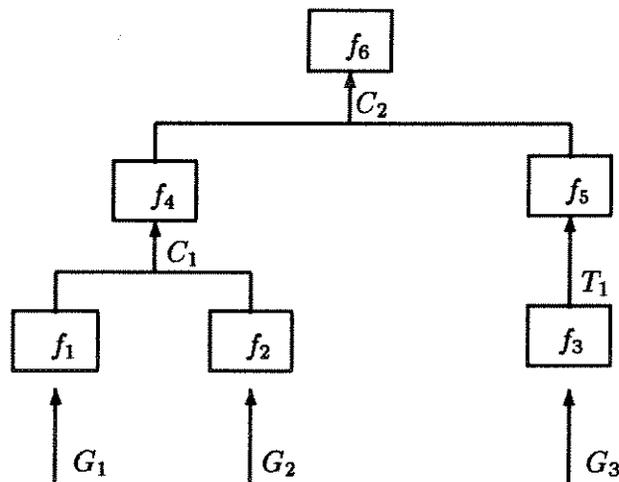


Figura 4.2: Estrutura hierárquica da textura

Exemplos típicos de funções geradoras são as primitivas de textura. Funções transformadoras podem ser filtros aplicados sobre uma textura original. E finalmente, combinação de texturas pode ser a composição de padrões ou de primitivas de texturas por exemplo nas diferentes formas da tabela 5.1.

4.3 Estrutura para um Sistema de Síntese de Texturas no ProSim

Um Sistema de Textura visaria o desenvolvimento de um pacote flexível para Síntese de Texturas e para tanto, deve permitir o acesso às funções de textura independentemente do algoritmo de *rendering* usado bem como a definição de texturas de maneira independente da resolução, forma e tamanho da imagem final.

O Sistema é proposto como possuindo dois ambientes: um independente ao processo de *rendering*, com funções para prototipar mapas de texturas através de chamadas às “primitivas de textura” (onde os mapas de texturas podem ser simples ou compostos) e um outro ambiente integrado ao processo de *rendering* para a aplicação da textura final aos objetos modelados por computador. A aplicação é feita através do relacionamento dos mapas da textura com os parâmetros de tonalização, relacionamento este definido pelas “funções de aplicação da textura” e as funções de “correspondência espacial”. O conjunto das funções para os diversos parâmetros caracteriza a “função completa de textura” que é na realidade a textura final do objeto. Sob estas considerações, os seguintes módulos seriam necessários:

1. Módulo de Primitivas de Textura;
2. Módulo de Ferramentas Gerais;
3. Módulo de Prototipação;
4. Módulo de Funções para Correspondência Espacial.
5. Módulo de Aplicação da Textura - interface com o *renderer*;
6. Módulo de Composição de Texturas;
7. Módulo de funções para Interação com o Usuário.

Como no ProSim a descrição de uma cena é feita através de um arquivo editor, no qual são usados comandos pré-definidos [Son90], na implementação de nossas experiências foram definidas palavras chaves a serem usadas neste arquivo para definir se o objeto possuirá textura, identificando a textura a ser usada, caso ela exista. Quando o programa que

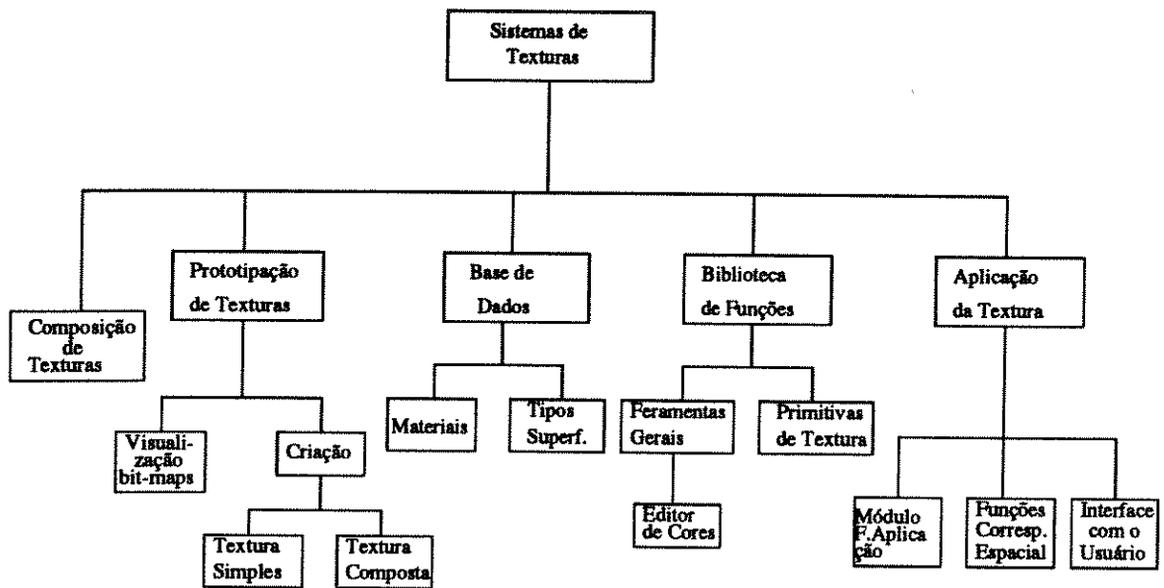


Figura 4.3: Estrutura para um Sistema de Síntese de Texturas

gera a síntese da cena é chamado, este analisa e interpreta os comandos e palavras chave de forma a gerar os processos necessários texturização. As texturas são identificadas por nomes pré-definidos numa fase prévia de prototipação de texturas. A Figura 4.4 mostra a comunicação do módulo de texturas com o ProSim.

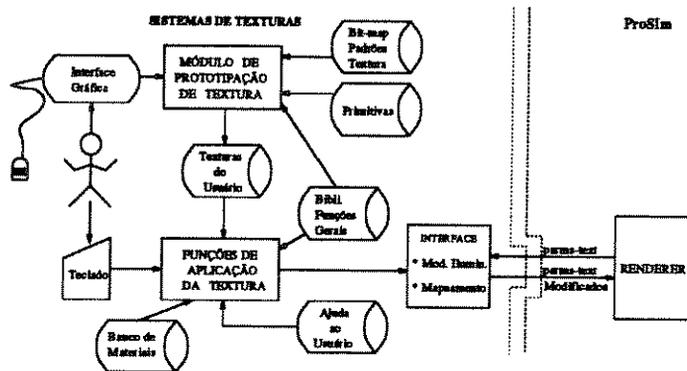


Figura 4.4: Diagrama geral do Sistema de Texturas no ProSim atual.

4.3.1 O Módulo de Primitivas de Textura

Consiste de uma biblioteca de algoritmos que implementam as funções parciais de textura (f_i) ou “Primitivas de Textura”. São chamadas assim porque são definidas com parâmetros de entrada que representam os fatores variáveis para a modelagem dos mapas de textura do usuário.

Um Mapa de Textura é o *frame-buffer* de uma imagem qualquer. pode ser modelado pela execução de primitivas de textura e descreve graficamente o comportamento de alguma característica óptica no espaço da textura. Estes mapas podem ser posteriormente interpretados como variações na intensidade de cor, brilho, transparência, etc, dependendo do parâmetro de tonalização ao qual sejam associados. Novas primitivas de textura poderão ser desenvolvidas via linguagem de programação (C) para acrescentá-las à lista de primitivas de textura a fim de serem usadas pelo usuário final. Algumas das primitivas já disponíveis serão discutidas no capítulo seguinte. No caso da função possuir parâmetros de chamada são definidos valores *default* para estes parâmetros.

4.3.2 Módulo de Ferramentas Gerais

Aqui estão contidas funções de uso geral usadas nos outros módulos do sistema, por exemplo, funções que implementam as transformações de rotação, escalamento, definição de janelas, antialiasing ou interpolação dos mapas de textura, etc.

Funções específicas deste módulo seriam funções que implementem filtros para destacar e apagar bordas, filtros para suavizar discontinuidades entre pixels, funções do tipo ruído, particularmente úteis para criar os mapas de textura das características brilho e rugosidade etc. A modelagem da direcionalidade de uma certa característica óptica pode ser feita com funções de rotação (implementadas no módulo de composição de padrões), funções como *bias* e *gain* citadas em [Per89] ou funções que mapeem um campo de textura em outro de geometria diferente para gerar o efeito de direcionalidade nos “texels”.

É muito importante que este ambiente possua um editor de cores pois mapas de textura podem ser gerados pela aplicação de primitivas a um espaço de cor. Neste caso o espaço de cor seria produzido interativamente através de funções que gerem novas

intensidades a partir da interpolação de duas cores básicas e que finalmente permitam gravar o valor numérico das cores geradas para seu uso durante a prototipação.

Este módulo deve também conter funções para consultar um banco de dados contendo os valores das constantes físicas que caracterizam cada material e/ou tipo de superfície. Também são necessárias neste módulo funções que controlem o valor final do parâmetro selecionado (ou seja após textura) de forma que por *default* o parâmetro não extrapole a faixa de variação permitida (isto é particularmente válido para modelos de iluminação com base física).

4.3.3 O Módulo de Prototipação

Este módulo é basicamente, um ambiente independente onde o usuário cria os mapas de textura. As funções principais são a **criação** de mapas pelo usuário, a **visualização** dos mapas de texturas criados e funções para **modificar**, **nomear** e **gravar** mapas de texturas.

Este módulo deve ter comunicação com o módulo de ferramentas para tratar os mapas obtidos até conseguir a aparência desejada. A idéia da prototipação é que o usuário possa obter os mapas de textura sem considerar ainda as variáveis que se introduzem devido ao *rendering*. O diagrama de blocos deste módulo aparece na Figura 4.5.

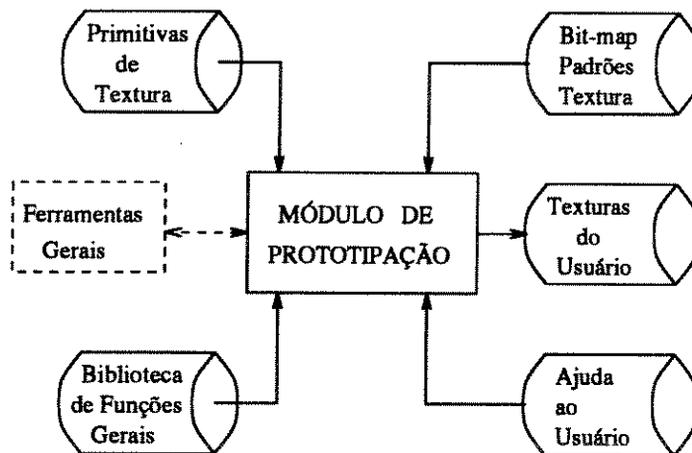


Figura 4.5: Diagrama de blocos do Módulo de Prototipação

A criação dos mapas de textura é feita pelo uso de primitivas de textura e de ferramentas de processamento de imagens.

O usuário poderá consultar via linha de comando a relação de primitivas de textura, as funções de transformação disponíveis (ferramentas) bem como os parâmetros das funções e seus valores *default* através de algum *help* associado.

Note-se que a prototipação de mapas de textura pode ser produzido de forma independente ao processo de *rendering* pois o mapa não é propriamente a textura do objeto. Tal como definido anteriormente, consideraremos o fenômeno “textura” somente na presença de um sistema de visão percebendo a interação da energia luminosa com um objeto material, ou seja, na definição das funções de aplicação da textura e sua execução desde o *renderer*.

4.3.4 Módulo de Funções para a Correspondência Espacial

Este módulo conterà diferentes funções de mapeamento para vários tipos de primitiva, mapeamento para funções de textura sólida e outras funções de apoio aos algoritmos de correspondência entre espaço objeto e espaço textura. Atualmente encontram-se implementadas no ProSim funções de mapeamento de relacionamento linear para as primitivas CSG do *raytracing*: esfera, cubo, cone, cilindro etc.. Novas funções podem ser definidas para objetos formados por *patches* ou para relacionamento entre variáveis objeto-textura diferente ao implementado. As funções de mapeamento são chamadas desde o *renderer* a fim de obter a localização (u, v, w) do espaço da textura, correspondente ao ponto $P(x, y, z)$ inversamente mapeado desde o espaço objeto.

4.3.5 Módulo de Aplicação da Textura

Uma vez definidos no módulo de prototipação os mapas de textura a serem usados, estes são aplicados aos objetos ou áreas de objetos através das funções de aplicação, definindo-se quais serão os parâmetros a serem “alterados” e de que forma será feita esta “alteração”. As variáveis (u, v, w) devolvidas desde o módulo de correspondência espacial são usadas para obter o valor da característica óptica em cada um dos mapas de textura, estes valores são passados às funções de aplicação para modificar os parâmetros de tonalização pertinentes a cada característica óptica.

Basicamente este módulo deve conter funções para:

- definir funções de aplicação e a “função completa de textura” ou textura final do objeto ou região;
- definir a que objetos ou regiões do objeto estará associada a textura final;
- definir cascatas de textura;
- associar os mapas de textura aos parâmetros de tonalização através das “funções de aplicação”;
- verificar que os limites de variabilidade de cada parâmetro sejam respeitados durante a aplicação da textura (válido para modelos de iluminação de base física);
- definir o palette de cores associado aos valores numéricos do mapa parcial de textura.

As funções de aplicação da textura devem ser conhecidas pelo *renderer* e para tanto definidas na descrição da cena.

4.3.6 Módulo de Composição de Texturas

Este módulo contém os submódulos de composição de padrões e a composição de primitivas (ou composição de procedimentos). Entenda-se por padrão o *frame-buffer* de qualquer imagem na tela.

No primeiro submódulo incluem-se funções que operam os valores dos pixels correspondentes entre um ou mais padrões ; funções para criar novos padrões, rotacioná-los e ordená-los colocando um acima do outro, um ao lado do outro, repetindo-os n vezes na direção de um eixo, etc.. Este módulo encontra-se já implementado no ProSim com uma interface texto para a composição.

A composição de primitivas ou procedimentos deve permitir a criação de novas primitivas pela composição de funções. Isto assemelha-se a um *pipeline* de procedimentos em que a saída de um é a entrada do outro. Deve conter funções de controle de fluxo e operadores de soma, produto, diferença etc. A composição de primitivas em nossa experiência foi feita via linguagem C e deve avaliar-se até que ponto deve-se definir uma nova interface sem perder a flexibilidade necessária que a linguagem de programação já possui.

4.3.7 Módulo de Funções para Interação com o Usuário

Este módulo deverá conter basicamente: uma interface interativa independente do *rendering* para a prototipação de mapas de textura e uma interface para a definição das funções completas de textura e as de aplicação de textura. A primeira das interfaces será chamada desde o módulo de prototipação de texturas. A segunda interface pode ser uma interface que grave um arquivo texto o qual após interpretação gerará dados a serem usados desde o *renderer*. Opcionalmente esta interface pode considerar a visualização da textura definida em uma cena padrão (cena de pouco custo de processamento).

4.4 O Processo de Texturização no ProSim

Em que consistiria então o processo de texturização no ProSim?

1. Durante a fase de modelagem do campo textural, consistiria em modelar mapas de textura para cada característica óptica reconhecida na textura. A modelagem destes mapas poderá ser feita via primitivas, composição de padrões, uso de ferramentas gerais ou arquivos de imagens. Esta fase pode ser implementada com interface interativa para o usuário e não precisa ter ligação com o *renderer*.
2. Durante a fase de aplicação da textura, consistiria em determinar para cada ponto visível da cena:
 - as coordenadas (u, v) do Espaço da Textura correspondente ao ponto $P(x, y, z)$ em tratamento. Este procedimento é feito desde o *renderer* por chamada aos procedimentos do módulo de correspondência espacial.
 - os valores das funções parciais de textura $f_i(u, v)$ obtidas dos i mapas de textura que modelam as i características ópticas reconhecidas na textura (ch_i).
 - o valor resultante da função completa de textura (ctf) obtido pelo uso das funções de aplicação $g_i(f_i)$ modificando os parâmetros do modelo de iluminação que modelam cada característica óptica ch_i . Um quadro dos p_i que modelam as características ópticas de brilho, cor, rugosidade e transparência na implementação atual do ProSim é mostrado na tabela 4.1; os parâmetros desta tabela

são os considerados no modelo de iluminação modificado de Whitted-Phong da equação 3.1.

devolver o controle ao renderer.

Tabela 4.1: Modelagem de características ópticas no modelo modificado de Whitted-Phong.

Característica óptica	Parâmetros que a modelam
Cor	K_a, K_d
Brilho	K_s, n
Transparência	K_t
Reflectância	K_r
Rugosidade	$P(x, y, z), \text{Normal}$

Quando todos os pontos visíveis do objeto tenham sido tratados pelo *renderer* o resultado obtido é o objeto texturizado. Uma ilustração dos passos necessários ao processo de texturização no ProSim é mostrada na figura 4.6.

Alguns dos elementos necessários à síntese de texturas no ProSim são enumerados a seguir para melhor ilustrar a implementação e operação da mesma:

Na Implantação do Sistema

1. Primitivas básicas de modelagem do campo de textura,
2. ferramentas de ajuda à modelagem do campo textural,
3. funções para a correspondência espacial do espaço 3D de definição do objeto e o espaço de definição da textura (mapeamento por exemplo),
4. uma estrutura que contenha os elementos variáveis do modelo de iluminação e do *renderer* que possam afetar a textura final,
5. ferramentas para a verificação da faixa de variação dos valores de cada parâmetro (valores coerentes ao modelo físico),
6. uma estrutura contendo as características ópticas reconhecíveis nas texturas em geral e que serão usadas como interface na descrição da textura,

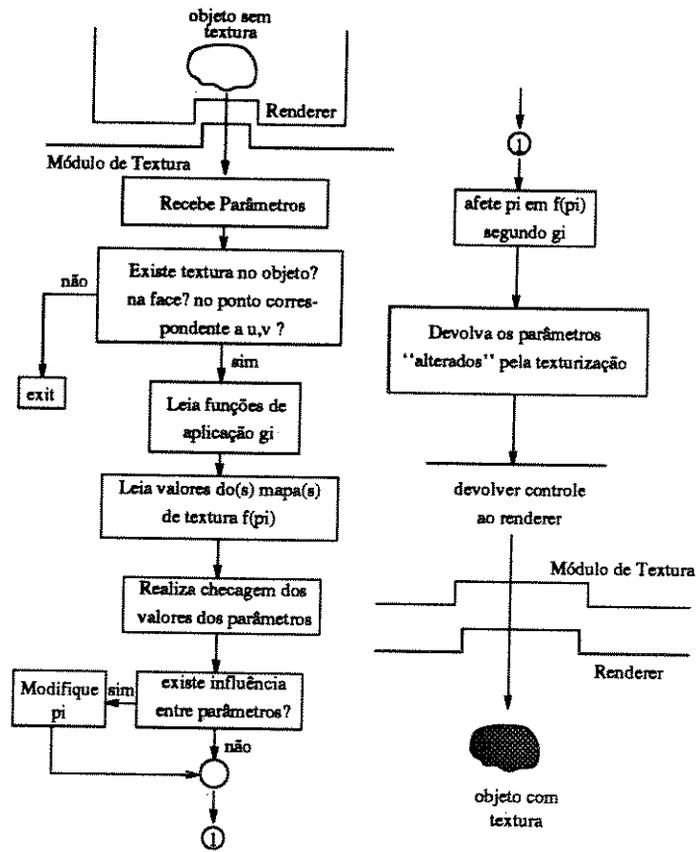


Figura 4.6:Fluxo de Procedimentos durante a texturização

7. uma estrutura contendo os parâmetros de tonalização que modelam cada característica óptica,
8. estabelecer procedimentos que verifiquem se existem relacionamentos entre parâmetros a serem verificados de forma a modelar coerentemente a característica óptica desejada (se este relacionamento existir).
9. uma interfaces amigável ao usuário contendo as constantes físicas correspondentes aos varios tipos de superfícies e materiais.

Ao modelar uma nova textura: Se não existirem texturas anteriormente modeladas que possam ser úteis para gerar a nova textura:

1. Identificar as características ópticas presentes na textura que se deseja simular,
2. se existirem banco de materiais ou de tipos de superfície identificar o apropriado para descrever a textura em termos dos mesmos,
3. para cada característica óptica identificada descrever o comportamento espacial da mesma (no caso de existir varios materiais descrever a posição e forma dos mesmos),
4. identificar as primitivas que possam ser úteis à modelagem do comportamento espacial desejado,
5. manipular as primitivas e ferramentas a fim de gerar mapas de textura que descrevam o comportamento desejado,
6. na aplicação da textura deve ser identificado o objeto ou áreas do objeto sobre as quais se deseja colocar textura
7. definição das funções de aplicação que serão usadas para definir a função completa de textura,
8. finalmente a escolha e uso da função de correspondência espacial deve ser realizada,
9. para cada ponto em tratamento obter o valor resultante da equação de iluminação quando são simultâneamente aplicados os mapas de textura aos parâmetros de tonalização.

10. caso não sejam alcançados resultados satisfatórios deve ser avaliada a modificação do código das primitivas existentes ou mesmo a criação de novas primitivas. Deve também ser avaliada a idoneidade do modelo de iluminação usado bem como a definição das funções de aplicação usadas para a modelagem do efeito desejado.

4.5 Exemplo: Como gerar uma textura realística de madeira

Mesmo quando pensamos em “madeira” existem vários tipos de madeira e várias aparências numa madeira tratada, portanto a forma em que essa textura pode ser modelada não é única. Neste caso tentaremos modelar uma textura que lembre uma tábua de madeira envernizada. Três são as características óticas que podemos identificar na textura observada: cor, brilho e rugosidade.

Percebe-se que a característica cor possui tons de marrom com linhas intermediárias de riscos pretos. A distribuição de cor varia entre elipses concêntricas com as bordas perturbadas e linhas entre elipses com direção quase paralela.

A característica brilho, no caso de uma madeira envernizada é uniforme na superfície toda. Duas abordagens podem ser usadas para sua simulação. No modelo empírico de Phong pode-se modelar como uma superfície de alta especularidade. No modelo de Cook e Torrance que considera a modelagem dos raios refletidos entre interfaces de diferentes materiais pode-se modelar as características de cada material (uma película transparente especular e uma superfície altamente difusora). Um brilho de características isotrópicas pode também ser observado mas seu efeito somente é notório na presença de movimento.

A característica rugosidade reduz-se a pequenos segmentos de linha orientados em direções quase paralelas sobre os quais acontecem leves depressões do nível da superfície. Os mapas de textura das características citadas estão representados na figura 4.7 a seguir.

O mapa da cor pode ser modelado tentando desenhar as regiões e assinalando cor a estas regiões. Uma outra abordagem é gerar diretamente um espaço de cor e introduzir funções de perturbação sobre este espaço até alcançar a geometria desejada.

O mapa de brilho não precisa de modelamento espacial pois o brilho é constan-

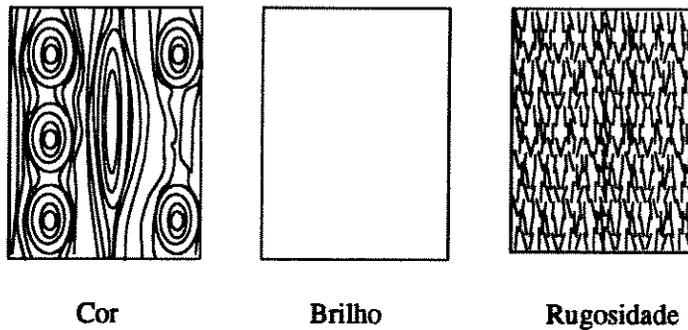


Figura 4.7: Mapas de textura para a modelagem da textura “táboa de madeira” de esquerda a direita: Mapa para cor, mapa para o brilho, mapa para a rugosidade.

te em todo o espaço da textura, somente é necessário definir valores para os parâmetros associados ao brilho.

O mapa de rugosidade pode ser obtido pelo posicionamento aleatório de segmentos de reta que apontam em direções quase paralelas.

O efeito final da textura simulada é mostrado na foto a seguir. A sub-imagem no canto superior esquerdo mostra a mesma cena antes da texturização para destacar o realismo acrescentado pela textura. Além da textura de madeira todas as texturas presentes na foto foram geradas seguindo a abordagem apresentada. A textura de tijolo possui um fator aleatório na característica “cor” e a textura de mármore no vaso foi gerada como textura sólida.

4.6 Observações e Comentários

Neste capítulo, considerando a base conceitual de 3.6.2 e os conceitos desenvolvidos em 4.2 foi apresentado um Modelo Geral para a texturização de objetos sintéticos pretendendo desta forma dar uma abordagem abrangente à síntese de texturas por computador. A proposta de implementação de um Sistema de Texturas esta orientada ao ProSim pois este foi o ambiente para nossas experiências. Neste capítulo interessou-nos apresentar as ferramentas para a solução do problema de texturização como um todo, desde a modelagem do campo textural até a aplicação da textura no objeto sintético. Detalhes dos algoritmos implementados são matéria do capítulo a seguir.

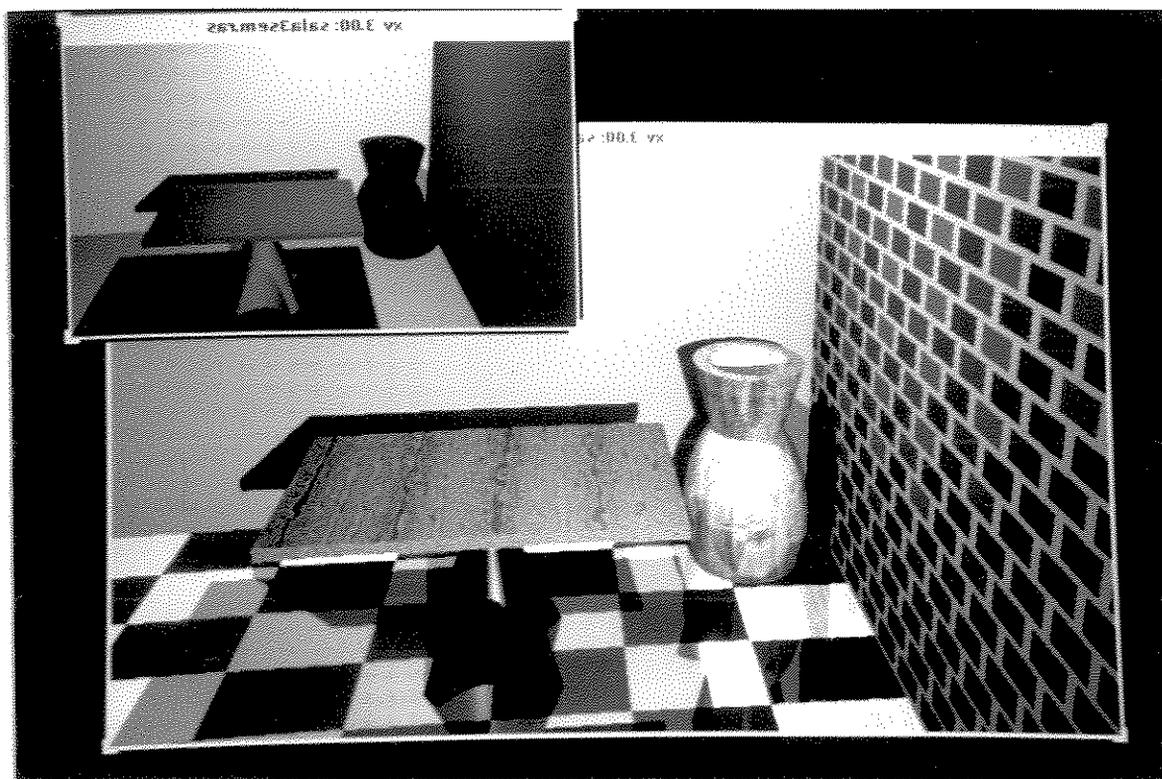


Foto 4.1: Uma cena sintética com a textura de madeira.

Capítulo 5

Recursos para a Síntese de Texturas

5.1 Introdução

Este capítulo discute alguns algoritmos correspondentes às fases de modelagem do campo textural e de aplicação de texturas às superfícies, implementados como recursos para a texturização no ProSim. Quanto à modelagem do campo textural, serão apresentados modelos estocásticos, determinísticos, os obtidos pela transformada inversa de Fourier e pela composição de modelos mais simples. Com relação à aplicação das texturas aos objetos modelados por computador são discutidos o mapeamento bidimensional, o mapeamento de texturas sólidas e a definição de algumas funções de aplicação de texturas.

Os algoritmos descritos neste capítulo foram desenvolvidos dentro do módulo de *ray-tracing* como uma experiência na síntese de texturas. Estes algoritmos estão inseridos nas facilidades de *rendering* em particular na versão de acesso público do ProSim que implementa: uma versão modificada do modelo de iluminação de Whitted-Phong [Mad91] para o *rendering*, um ambiente de modelagem de sólidos por CSG e o modelo câmera-observador especificado em [Gla88]. O sistema trabalha nas estações SUN, sistema operacional UNIX e está desenvolvido em linguagem "C".

São também apresentadas imagens, exemplo de mapas de textura e de objetos texturizados mostrando-se como é possível obter versatilidade na implementação apresentada a partir de algumas poucas primitivas básicas.

5.2 Fase de Modelagem da Textura: Algoritmos de Geração

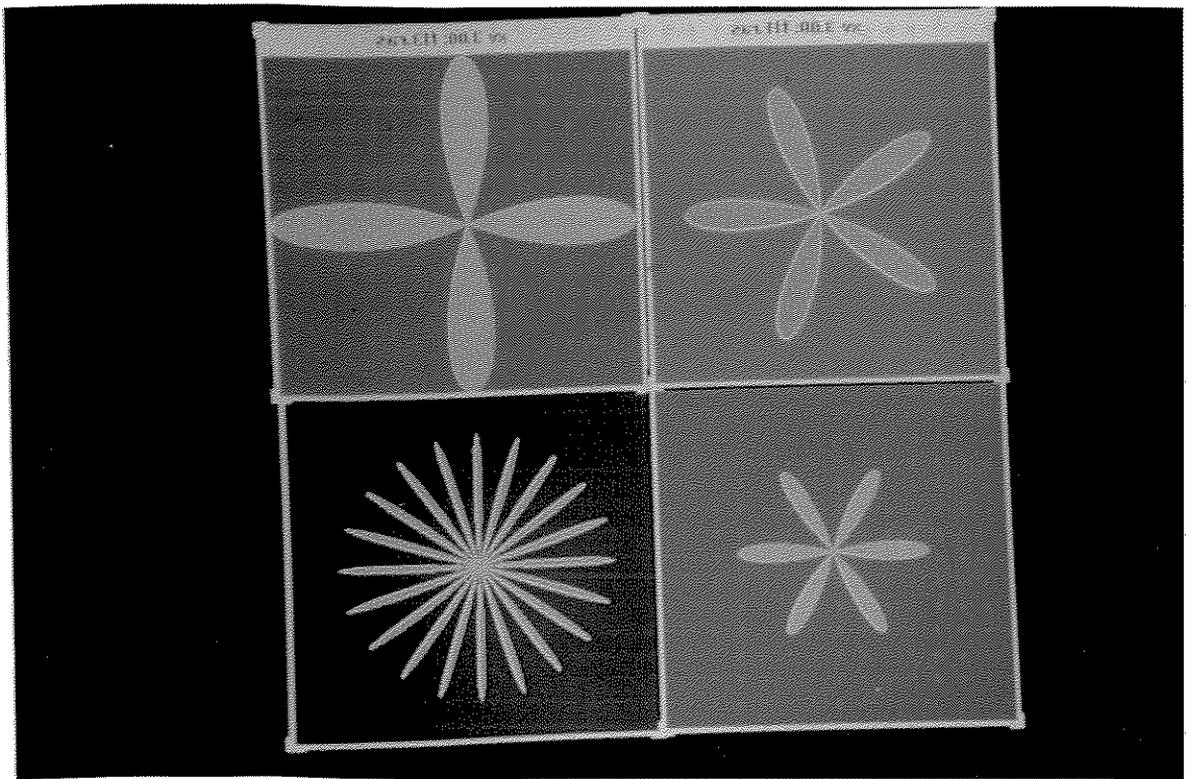


Figura 5.1: *t-flower* em diferentes instâncias para diferentes valores de cor, número de pétalas e tamanho da pétala

5.2 Fase de Modelagem da Textura: Algoritmos de Geração

A modelagem da textura, segundo a abordagem proposta, consiste na verdade em modelar o comportamento das características ópticas observadas na textura, modelagem esta feita através de primitivas que como resultado geram os Mapas de Textura. Uma vez obtidos os mapas de textura, estes são simultaneamente aplicados em cada ponto durante o *rendering* a fim de representar a textura total. Os algoritmos que implementam os “mapas de textura” foram programados como primitivas que podem ser executadas desde a linha de comando. A visualização do mapa gerado pode ser feita na tela. Cada primitiva possui parâmetros de chamada que permitem a geração de instâncias diferentes da mesma primitiva e valores *default* a serem usados caso não sejam fornecidos parâmetros específicos de instanciação. Por exemplo na Foto 5.1 mostram-se mapas de cor correspondentes a instâncias diferentes da primitiva *t-flower*, onde se podem observar os efeitos de variar os parâmetros: cor de fundo, cor da flor, numero de pétalas, tamanho de pétala.

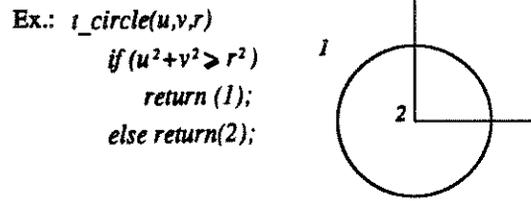


Figura 5.2:Pseudo-código da primitiva *t_circle*

Também é possível alterar o tamanho da imagem resultante, embora o efeito não seja mostrado na foto.

Estratégia idêntica é adotada na implementação de outras primitivas. São previstas duas modalidades de execução de cada primitiva: desde a linha de comando e desde o *renderer*. No primeiro caso a saída é o *frame-buffer* do mapa de textura modelado, o qual é gravado em arquivos com extensão .r, .g e .b.

5.2.1 Modelos Determinísticos

Estes algoritmos implementam: funções geométricas bem conhecidas e arquivos de imagens ou tabelas de dados a serem utilizadas como procedimento gerador de mapas de textura.

5.2.1.1 Funções Geométricas

Tratam-se de primitivas de textura, que fazem uso de funções matemáticas para modelar comportamentos no espaço da textura. Como resultado disto o Espaço da Textura fica subdividido em subregiões. Quando executada desde a linha de comando a função retorna uma imagem. Quando chamada desde o *renderer* a função cria subregiões, onde a cada subregião lhe é atribuído um identificador que será associado a valores de intensidade. Por exemplo, na Figura 5.2 um círculo ou uma esfera podem ser usados para criar uma primitiva textural, onde a região fora do círculo estaria identificada como 1 e a região dentro do círculo como 2. Quando a primitiva é executada desde o *renderer* devolve a identificação da subregião atingida pelo ponto inversamente mapeado.

Por convenção, as subregiões serão identificadas da esquerda para a direita e de

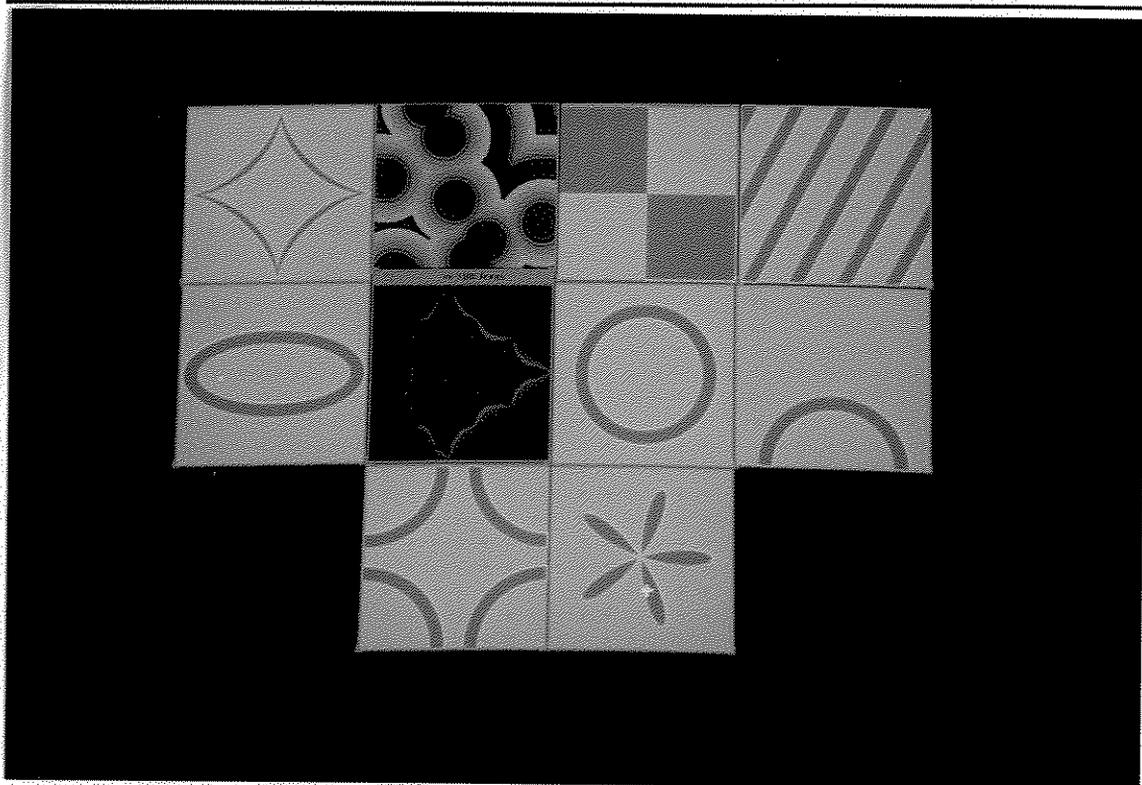


Figura 5.3: Algumas das primitivas geométricas implementadas.

cima para baixo, sendo o centro de coordenadas o centro do espaço textural, que no espaço unitário corresponde às coordenadas $(0.5,0.5)$, ou o canto inferior esquerdo, que no espaço unitário corresponde às coordenadas $(0,1)$.

Em geral, as funções geométricas implementadas são facilmente definidas nos espaços de duas ou três dimensões. Uma breve descrição das primitivas geométricas aparecem no anexo A. Algumas primitivas geométricas aparecem na Foto 5.3.

5.2.1.2 Bit-map de arquivos de imagens

Uma imagem bi ou tri-dimensional é levada ao espaço unitário de textura e o valor da intensidade em cada pixel é modulada a valores dentro do intervalo $[0, 1]$ (isto porque, geralmente o valor do modelo de textura obtido é usado diretamente como valor para os parâmetros do modelo de iluminação).

Quando a imagem é usada para alterar a normal à superfície, áreas mais obscuras

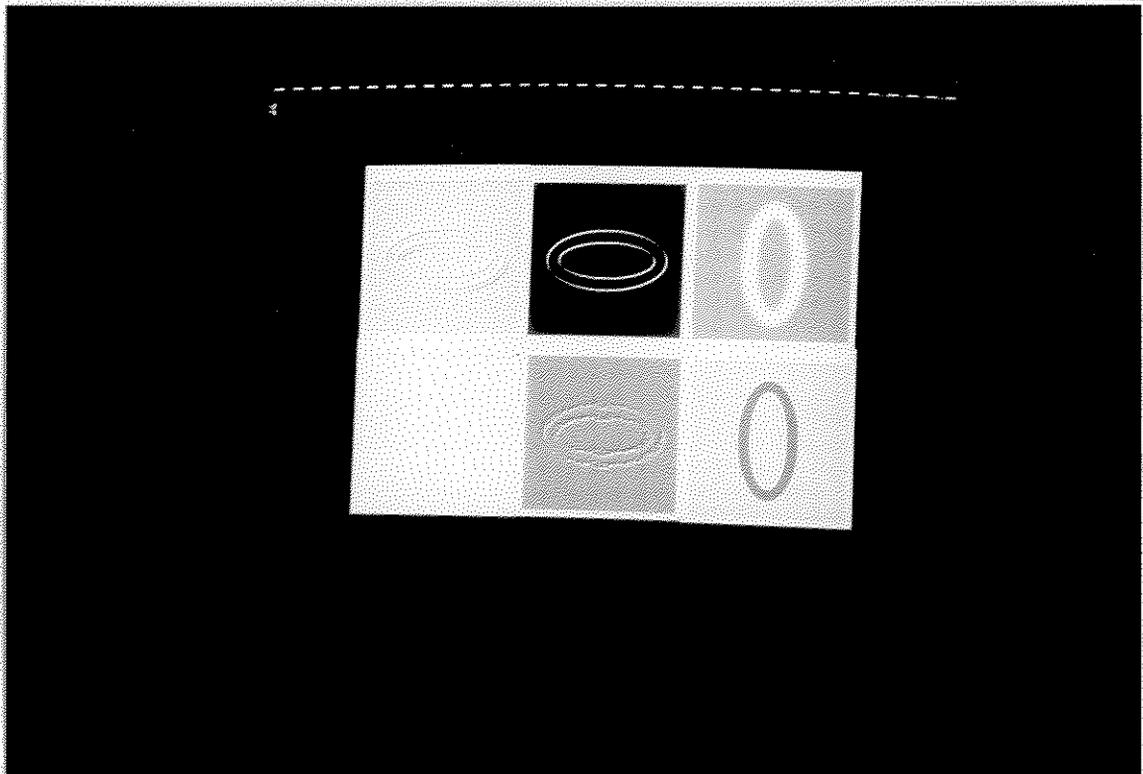


Figura 5.4:Diversos mapas gerados a partir da primitiva elipse.

são interpretadas como depressões e as mais claras como elevações. A imagem original é previamente filtrada a fim de obter o gradiente de intensidades da imagem onde a magnitude do gradiente expressa a diferença local de intensidades entre pixel vizinhos e a direção do gradiente o sentido e crescimento da intensidade luminosa.

Diversas ferramentas podem ser usadas para criar mapas de textura a partir de uma imagem. Na Foto 5.4 aparecem diversas imagens geradas pela aplicação de filtros sobre a imagem de uma instância de primitiva de elipse.

5.2.1.3 Funções de Projeção

Esta é uma técnica em que texturas tri-dimensionais são obtidas pela projeção de texturas bi-dimensionais ao longo de um caminho (*path*). Assim por exemplo, uma textura bi-dimensional $\rho(x, y)$ pode gerar uma textura 3D utilizando uma função de projeção ortogonal R , onde:

$$R(x, y, z) = \begin{cases} \rho(x, y) , & \text{para } x \text{ e } y \in [0, 1] \\ 0 , & \text{nos outros casos.} \end{cases}$$

Neste caso R seria simplesmente a projeção da textura ρ ao longo do eixo Z . Isto quer dizer que cada elemento discreto da textura ρ , gera um paralelepípedo que se estende nos sentidos positivo e negativo, paralelo ao eixo Z .

Um recurso muito importante é o fato de que R pode ser rotacionado, transladado ou escalado com relação ao objeto que está sendo texturizado, de tal forma que as direções de projeção, em última instância, não necessariamente coincidam com o eixo Z .

Projeções ortogonais seriam uma variedade entre as diversas possibilidades de projeção. Para se obter texturas mais interessantes, podem ser usadas técnicas de projeção cilíndrica ou caminhos de projeção diferentes de linhas retas; por exemplo, um caminho ondulado. Outra possibilidade é definir a textura sólida como o resultado da interseção no espaço de duas texturas bi-dimensionais; por exemplo desde os planos x - y e y - z .

5.2.2 Modelos Estocásticos de Textura

Quando a imagem gerada pela primitiva possui uma geometria complexa (caso das primitivas estocásticas e funções fractais) a chamada à primitiva gera um conjunto de valores que é gravado em um arquivo com extensão *.pic*. Posteriormente estes resultados são associados a alguma paleta de cores, estabelecendo-se um mapeamento entre os valores armazenados e o domínio da paleta. Neste mapeamento é imprescindível definir a faixa de valores que a função pode assumir e para isto a função é testada previamente.

Também as primitivas estocásticas são definidas com parâmetros de entrada de forma a obter versatilidade. Os parâmetros mais usados são:

- Níveis de recursão;
- Semente para a função randômica;
- Identificador para o número de iterações ;
- Constantes das funções fractais, etc.

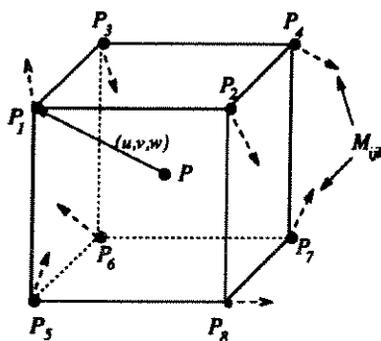


Figura 5.5: Pontos vizinhos ao ponto $P(x, y, z)$.

5.2.2.1 A Função t_Noise

Esta primitiva para modelos de texturas foi implementada como em [Per89] e é definida como uma função que contribui com um valor aleatório a qualquer organização regular sem perda da continuidade.

Os valores devolvidos por esta função variam entre 0 e 1 para qualquer ponto P de coordenadas x , y e z e estão sendo implementados de maneira que dois pontos suficientemente diferentes geram valores independentes de t_noise . Entre outras características da função, ela é contínua para os 3 eixos de coordenadas e, tal como indicado por Perlin [Per89], tem mostrado invariância estatística na rotação e translação.

A função considera os seguintes elementos:

- um ponto P de coordenadas $P(x, y, z)$;
- a vizinhança de $P(x, y, z)$ no espaço, ou seja os pontos P_1, \dots, P_8 (ver Figura 5.5);
- um conjunto de gradientes M_{ijk} pseudo-radômicos em cada um destes pontos;
- um conjunto de 8 vetores posição $(u, v, w)_{ijk}$, definidos como $P - P_l$, $l = 1, \dots, 8$.

Logo:

$$(u, v, w)_{ijk} = (x - i, y - j, z - k);$$

- Ω_{ijk} como uma variável pseudo-aleatória associada a cada ponto vértice, dado pelo produto escalar dos vetores M_{ijk} e $(u, v, w)_{ijk}$, ponderado pelos valores $\omega(u)$, $\omega(v)$ e $\omega(w)$.

$$\Omega_{ijk} = \omega(u) \cdot \omega(v) \cdot \omega(w) (M_{ijk} \cdot (u, v, w)_{ijk}) \quad (5.1)$$

onde

$$\omega(\phi) = \begin{cases} 2\|\phi\|^3 - 3\|\phi\|^2 + 1 & , \text{ se } \|\phi\| < 1 \\ 0 & , \text{ em outros casos} \end{cases} \quad (5.2)$$

Tem-se então

$$t.noise(x,y,z) = \sum_{i=i_{menor}}^{i_{maior}} \sum_{j=j_{menor}}^{j_{maior}} \sum_{k=k_{menor}}^{k_{maior}} \Omega_{ijk}(x-i)(y-j)(z-k) \quad (5.3)$$

A Função *t.noise* procura atribuir valores aleatórios aos pontos do espaço, porém, como é desejada uma continuidade entre os diversos pontos, em vez de gerar-se diretamente valores aleatórios a cada posição, são gerados os vetores *F* e *G*, ambos com o mesmo número de elementos *n*, sendo que:

- o vetor *F*, de índices para *G*, consiste dos *n* primeiros números inteiros desordenados aleatoriamente;
- cada elemento do vetor *G* é uma tripla, que representa um vetor gradiente correspondente a uma amostra de *n* pontos, uniformemente distribuídos sobre a superfície de uma esfera imaginária de raio $r = 1$. Desta maneira assegura-se o comportamento mais ou menos contínuo do gradiente para pontos vizinhos.

Devido à natureza aleatória da função *t.noise* é importante evitar que seja atribuído o mesmo gradiente (uma mesma tripla de *G*) a pontos $P(i, j, k)$ vizinhos. Para tal, as coordenadas *i*, *j* e *k* são introduzidas como elemento de distinção na determinação do índice de *G* a partir do vetor *F*, segundo o seguinte procedimento:

$$G(i, j, k) = G[F(i + F(j + F(k \bmod n) \bmod n) \bmod n)]$$

onde $a \bmod b$ representa o resto da divisão de a por b .

Assim, se $n = 12$, $F = [10, 3, 7, 0, 4, 9, 2, 8, 6, 11, 1, 5]$

tem-se que, para os pontos vizinhos,

$P(i, j, k) = (50, 24, 10)$ é escolhido $G[9]$

$P(i, j, k) = (51, 24, 10)$ é escolhido $G[2]$

assegurando com isso que o gradiente para um ponto dado seja sempre o mesmo e que, ao mesmo tempo, para dois pontos vizinhos sejam distintos.

Com este procedimento é definido o conjunto de vetores gradientes (M_{ijk}) correspondentes aos n pontos P_{ijk} vizinhos a $P(x, y, z)$.

O vetor gradiente resultante para $P(x, y, z)$ é o valor da função t_noise nesse ponto e é definida como na Equação (5.3), a partir dos vetores obtidos nas Equações (5.1) e (5.2).

Note que se n for maior, a continuidade obtida será também maior, sendo $n = 8$ um valor razoável para os cálculos.

5.2.2.2 Perturbação de padrões regulares

Qualquer função que produza padrões regulares pode ser perturbada pela adição de um fator aleatório de irregularidade. Um exemplo simples que produz uma textura bem comportada tridimensional é:

$$T(x, y, z) = \text{Factsin}\{2\pi[Ax + B/10(C \text{ random}(x, y, z) + 1 + \cos(2\pi Dy))]\}$$

onde A , B , C e D são constantes quaisquer que dão variedade ao padrão.

5.2.2.3 A Função `t_d_noise`

Esta função é a derivada da função `t_noise` e é útil para ser aplicada diretamente quando o parâmetro que será afetado pela função de textura é a normal às superfícies, onde `t_d_noise` é a taxa instantânea de variação de `t_noise` ao longo do eixos x , y e z . Lembrar que, quando se deseja alterar a forma de uma superfície S por uma função $f(\cdot)$, a simulação visual deste efeito é conseguida alterando-se a normal à superfície, segundo a derivada da função de perturbação, como discuto em 2.2.3.

$$t_d_Noise = \frac{T_Noise_p - t_Noise_{p-vizinho}}{distancia - ao - vizinho}$$

5.2.2.4 A Função `t_turbulence()`

Esta função consiste em uma implementação simples, similar à proposta por Perlin, baseada no modelo para síntese de turbulência, desenvolvido em [Per89]. O modelo de turbulência simplificado é representado por,

$$turbulence(x) = \sum_i \left\| \frac{1}{2^i} Noise(2^i x) \right\|$$

Neste caso, uma frequência de $\frac{1}{2^i}$ é uma boa aproximação para representar a turbulência (como no parâmetro de auto-similaridade). A fórmula geral de turbulência é:

$$turbulence(x) = \sum \left\| Noise(x * \frac{1}{f}) * f \right\|$$

sendo f a frequência da função que representa a turbulência e $\frac{1}{f}$ a amplitude da mesma.

Em termos práticos, o algoritmo de implementação é baseado num *loop*:

```

Turbulence(p){
  s = 1;
  for(i = 1; i = valor; i++){
    t = valor_absoluto(1/s * Noise(p * s));
    s = s * 2;
  }
  return t;
}

```

onde s são os valores da amplitude em cada escala ($escala = \frac{1}{f}$) e i é o nível de recursão da turbulência (quanto maior for i , maior será o detalhamento da função).

Acrescentamos, para uma melhor compreensão, a explanação dada por Perlin à expressão $Noise(\frac{P}{scale}) * scale$: "...pois nela está expressado que, a cada escala, a quantidade de perturbação adicionada é proporcional à magnitude da escala, o qual cria a característica de auto-similaridade, ou padrão, do tipo $\frac{1}{f}$ ". Por outro lado, a função *valor_absoluto*, em cada iteração, assegura que o gradiente terá bordas descontínuas em todas as escalas, o que dará a impressão de descontinuidade, própria de fluxos turbulentos. A Foto 5.6 mostra uma

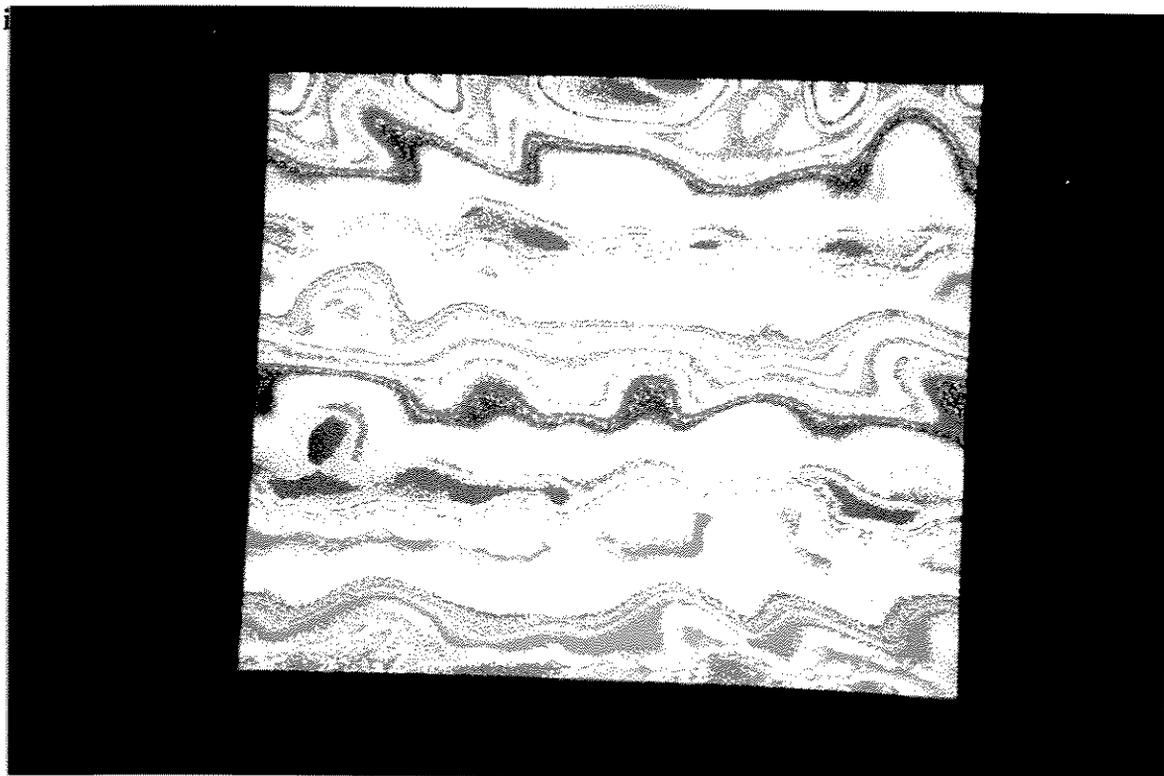


Figura 5.6: Instância da Função *t.turbulence*.

5.2.2.5 A Função *t.bubble()*

Existem dois enfoques para este modelo:

- criar a textura 3D no espaço posicionando aleatoriamente bolhas com tamanhos e formas diferentes. Através de uma função randômica os diâmetros de uma primitiva

esfera são variados (no caso de valores distintos para diâmetro transversal a esfera se converterá num elipsóide);

- um segundo enfoque cria a textura no espaço 2D. O algoritmo que produz esta textura é o usado para determinar qual dentre um conjunto de pontos é o mais próximo de um ponto dado.

A função implementada gera aleatoriamente N pontos sobre o plano da textura. Se as distâncias calculadas são representadas por cores, então em volta de cada um dos pontos gerados criam-se círculos concêntricos, cujos tons de cor variam segundo a distância ao centro.

Se a gama de cores mapeada for desde o branco até o preto, passando pelos diversos tons de cinza e esta ordem guardar correspondência com as distâncias de cada um dos pontos ao ponto de referência (cores mais claras para as distâncias menores), então a impressão final da textura é de esferas apertadas umas contra as outras (a variação de cor cria impressão 3D), o que fará lembrar a aparência de bolhas umas junto às outras. O princípio é o mesmo ao usado na geração do diagrama de Voronoi (o qual é uma malha que representa as linhas de equidistância entre cada par de pontos dentre um conjunto de pontos distribuídos aleatoriamente no espaço).

Como o algoritmo baseia-se no cálculo da distância, se este cálculo for estendido para três dimensões, então a função pode ser aplicada em texturas sólidas também.

5.2.2.6 Modelo de buracos

Cria a impressão de buracos esféricos aleatoriamente posicionados. A função de textura no espaço 3D está definida por:

$$T(x, y, z) = -\text{Fact}\{e^{-N^2R/D}\}$$

onde $R = +\sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2}$ $N = 4/D$ e D é o diâmetro da esfera que simula o buraco.

5.2.2.7 Modelos para variações de cor

Estes modelos servem para variar tanto cor quanto transparência, brilho ou outras características das superfícies. Considerando somente a cor são possíveis três enfoques:

- a variação de cor está relacionada com as coordenadas originais;
- a variação de cor está relacionada com as coordenada atuais do ponto;
- a variação de cor é dependente do espaço da textura.

Uma função interessante para cor é definida como segue:

$COLOR.R := Fact \ COLOR.R$

$COLOR.G := Fact \ COLOR.G$

$COLOR.B := Fact \ COLOR.B$

Onde Fact é definido como:

$$Fact := 0.5 + 0.5 \left\{ \sum C_i (\sin(w_i x + P_z) + 1) \sum C_i (\sin(w_i z + P_x) + 1) \right\}$$

com $C_{i+1} = C_i / \sqrt{2}$, $w_{i+1} = 2w_i$, $P_z = K_1 \sin(2\pi z)$ e $P_x = K_2 \sin(2\pi Bx)$

onde a regularidade da função de cor ¹ é controlada pelos fatores K_1 e K_2 .

5.2.2.8 As funções t_fracxx()

Existem fenômenos não periódicos e caóticos que possuem uma ordem bem definida por propriedades chamadas de auto-semelhança e auto-similaridade. As funções que modelam este tipo de comportamento foram chamadas de fractais por Mandelbroot e são obtidas pela iteração de funções cujo domínio provém de números complexos ou pela repetição de procedimentos matemáticos com as seguintes propriedades:

- Possui infinitos detalhes a qualquer escala,
- é irregular,

¹Esta classe de funções também pode ser usado para representar terra como em [Gar85]

- possui frequentemente alguma forma de auto-semelhaça.

Estas funções são um recurso muito importante para a modelagem de certas formas e texturas deste tipo. Mandelbrot propõe o Movimento Browniano Fraccional como um recurso para modelamento de superfícies terrestres, assunto já comentado no Capítulo 2.3.2.

Atualmente os fractais são definidos na comunidade de Computação Gráfica como objetos que possuem algum grau de autosimilaridade seja esta exata ou estatística². Isto amplia a definição original de Mandelbrot.

Na caracterização matemática do fractal, diz-se que possui a propriedade de auto-similaridade e dimensão fracionária. O conceito de **auto-similaridade** faz referência à propriedade pela qual o objeto como um todo é similar a uma porção do mesmo sendo que a palavra “similar” indica que o objeto não é exatamente igual pois podem existir diferenças devido a rotação, translação, escalamento etc.

A dimensão de um fractal não é satisfatoriamente caracterizada pelos valores tradicionais de 1D ou 2D e assim são usados valores fracionários no intervalo $< 1, 2 >$ para defini-la. Isto é chamado de **dimensão fracionária**, a qual é sempre maior ou igual a sua dimensão topológica. Uma maneira de entendê-la é dizer que avalia quanto espaço ocupa a curva que representa a função, isto pode ser explicado fazendo uso de um fractal bastante conhecido: a curva de Koch. Quando uma linha é dividida em N partes, cada uma mede: N/N o que pode ser expresso como a linha original inversamente escalada pelo fator: $N^{1/1}$. Se o que dividimos for um quadrado, cada novo lado dos N quadrados gerados pode ser expresso como o lado original inversamente escalado por $N^{1/2}$. Se observamos a curva de Koch (Figura 5.7) vê-se que inicialmente possui uma longitude de $N=4$ e que após a divisão cada parte linear mede $1/3$ da original. Seguindo a forma de fator de escala inversa, a dimensão d de cada nova parte linear pode ser escrita como $N^{1/d}$ ou seja $4^{1/d}$ o que é igual a 3, de onde resolvendo par d verifica-se que $d = 1.26$, ou seja a dimensão d assumiu um valor fracionário.

Os dois objetos mais famosos entre os fractais são o *Julia-Fatou set* e o *Mandelbrot set*. Ambos originados a partir da equação:

²Geralmente a definição matemática altamente precisa, própria dos fractais conhecidos implica em autosimilaridade estatística em todas as resoluções.



Figura 5.7: Curva de Koch.

$$x \rightarrow x^2 + C$$

onde x é um número complexo.

Quando $|x| = 1$ se obtém o *Julia-Fatou set* e para todos os possíveis valores de C que dão um *Julia set* conectado obtém-se o *Mandelbrot set*. Um fractal é chamado determinístico quando é obtido num processo recursivo ou iterativo se faz uso de elementos aleatórios (o mesmo processo gerará sempre o mesmo fractal). Num fractal randômico incorpora-se ao processo de geração aspectos aleatórios que influenciam a aparência final do fractal gerado.

Neste trabalho apresentam-se fractais geométricos e randômicos. Um fractal geométrico repete exatamente o mesmo padrão em todas as escalas, um fractal randômico pode parecer ao geométrico mas os detalhes não são perfeitamente idênticos em todas as escalas.

5.2.3 Modelagem a partir da Transformada de Fourier

A modelagem de certas texturas é sumamente complexa quando tratada no domínio espacial, mas quando vista no domínio da frequência o espectro resultante é bem mais simples e, portanto, mais susceptível de ser modelado.

Quando não se dispõe do modelo da textura final e sim de sua transformada de Fourier, este algoritmo implementa a obtenção da textura a partir da aplicação da transformada inversa de Fourier (FFT^{-1}). Isto porque, tal como levantado por Lewis [Lew84], em alguns casos, trabalhar no espaço das frequências facilita a descrição das texturas. As experiências obtidas com base na plataforma KHOROS de domínio público foram feitas apenas para a obtenção de campos de textura, ficando ainda por explorar a modelagem da representação da transformada no espaço das frequências. A modelagem da textura no espaço das frequências pode ser obtida através de algum aplicativo de desenho por computador. Para conseguir relacionar a textura desejada com a imagem correspondente no espaço das frequências foram obtidas as transformadas de texturas conhecidas de difícil representação e analisados os espectros resultantes.

5.2.4 Composição de Texturas

É possível desenvolver variedades de texturas a partir de texturas mais simples (funções que criam primitivas de texturas).

Duas são as abordagens para compor texturas: a composição de texturas descritas proceduralmente e a composição de padrões de textura ou *texels*³ Em nossa experiência a composição de primitivas de textura foi feita via linguagem de programação "C". A composição de padrões foi feita através de um modulo independente com interface textual.

³de *texture element*, esta definido como uma primitiva visual com certas propriedades invariantes que se repete dentro de determinada área em diferentes posições, deformações e orientações.

5.2.4.1 Composição de Primitivas de Textura

Podem ser primitivas determinísticas ou estocásticas. Os mapas de textura são definidos por chamada à primitiva textural com parâmetros apropriados e a composição é feita usando estes “mapas” das seguintes maneiras:

- Pela repetição do mesmo padrão n_u e n_v vezes ao longo dos eixos u e v , do espaço de textura (simulando texels);
- pela execução simultânea das primitivas operando-se no mesmo ponto (u, v) os valores retornados por cada primitiva. Exemplo: $t_composta = t_fractal + t_circle$;
- pela execução alternativa das primitivas no espaço da textura, gerando-se por exemplo um ordenamento particular dos mapas de textura numa matriz de $n \times m$ elementos. Cada elemento da matriz é um mapa de textura descrito no espaço unitário. Os critérios de ordenamento dos mapas de textura podem ser descritos através de funções matemáticas tal como:

```
begin diagonal_flower
  if( $u'=v'$ )
    texture = t_circle
  else
    texture = t_flower;
end diagonal_flower
```

onde u' , v' são os índices da matriz definidos como $u' = \text{int}(u * n_u)$ e $v' = \text{int}(v * n_v)$ quando u, v são as variáveis de acesso no espaço unitário de definição de cada modelo de textura e n_u, n_v representam o número de elementos da matriz nas direções u, v respectivamente. A figura 5.8 mostra o comportamento da matriz de modelos de textura para *diagonal_flower*;

- pela execução em *pipeline* das primitivas, gerando-se uma “cascata de mapas de textura”. Nesta abordagem o usuário é capaz de identificar as subregiões contidas em cada mapa e preencher cada subregião com o mapa gerado por uma outra primitiva se assim se desejar. A definição da cascata é feita através de um procedimento do tipo:

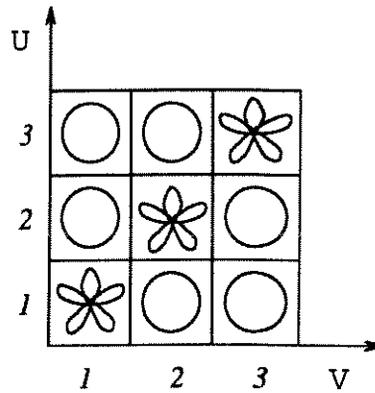


Figura 5.8: Matriz de modelos de textura de *diagonal_flower*

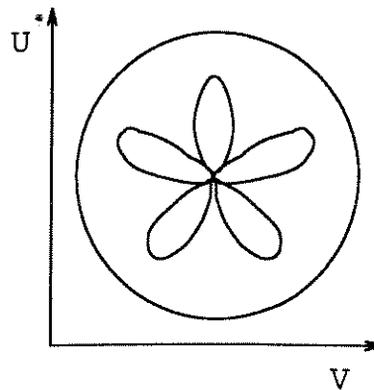


Figura 5.9: Cascata de Textura para *cascade_circle_flower*

```

begin cascade_circle_flower
texture = t_circle(u,v)
  if(t_circle(u,v)=1)
    texture = t_flower;
  end cascade_circle_flower

```

A Figura 5.9 mostra uma cascata de textura para o procedimento *cascade_circle_flower*.

A Foto 5.10 mostra um exemplo de uma composição da primitiva *t_lista* e a função de turbulência.

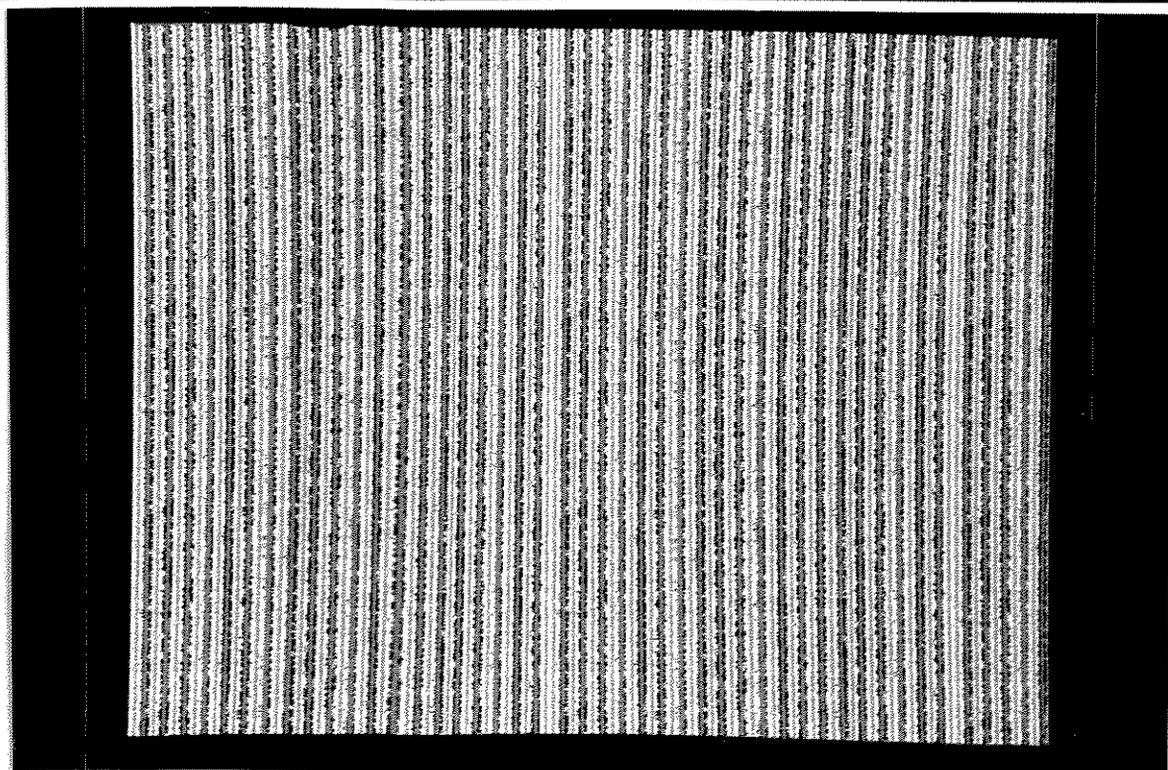


Figura 5.10: Função *t.turbulence* composto com a primitiva *t.lista*.

A composição pode também ser feita durante a aplicação da textura, pela criação de um procedimento que avalie em cada ponto tratado pelo *renderer* o valor de algum parâmetro de interesse do modelo de iluminação e escolha a partir do resultado o mapa de textura a ser usado. Isto gera efeitos valiosos na manipulação do espaço-objeto, por exemplo: a texturização de porções da superfície do objeto, a simulação de mais de um material n textura e até efeitos inéditos como a simulação da “resolução” da textura e aparências anisotrópicas. As diversas formas de compor primitivas de textura durante a modelagem e aplicação das texturas assim como os resultados possíveis de serem obtidos são mostrados na tabela 5.1.

5.2.4.2 Composição de Padrões de Textura

Nesta abordagem chama-se de padrão a uma imagem qualquer. Um novo modelo de textura pode ser gerado a partir da manipulação de padrões, considerando os elementos a seguir:

+_up (*P1,P2*) Define um só padrão com *P2* sobre *P1*.

+_down (*P1,P2*) Define um novo padrão com *P2* abaixo de *P1*.

AND (*P1,P2*) O novo padrão é composto a partir do resultado da operação AND lógico entre o conteúdo dos *frame-buffer* de *P1* e *P2*.

OR (*P1,P2*) O novo padrão é composto do resultado do OR lógico entre o conteúdo binário dos padrões *P1* e *P2*.

XOR (*P1,P2*) O novo padrão é composto do resultado do XOR lógico entre o conteúdo binário dos padrões *P1* e *P2*.

As operações de transformação seriam:

Rotate (α , *P*) Rotacionar o padrão *P* em relação ao centro do ângulo α especificado em radianos.

Scale($\Delta x, \Delta y$, *P*) O padrão *P* é escalado às taxas Δx para o eixo *x* e Δy para o eixo *y*.

Filter(*nom_filtr*, *P*) É aplicado o filtro especificado no primeiro argumento ao padrão *P*.

Shift ($\Delta x, \Delta y$, *P*) O novo padrão *P* é obtido deslocando o centro de coordenadas do padrão em $\Delta x, \Delta y$.

P1 = P Atribue o que estiver à direita do = ao padrão especificado na esquerda.

Um exemplo da sintaxe do arquivo de composição de padrões é mostrado a continuação, a sintaxe esta definida em [Fra92].

```
tesselation teste2;
```

```
size 400 by 400;
```

5.3 Fase de Modelagem da Textura: Algoritmos de Geração

```
tessellation retangular;

celula 200 by 200;

images

imagem1 = leo 200 by 200;
imagem4;
imagem5;
imagem6;
imagem7;

end images;

imagem4 = expand imagem1 to 300 by 300 background 0,0;
imagem5 = rotate imagem4 by 20 around 150, 150;
imagem6 = clipping imagem5 from 50,50 to 250,250;

built with
(1,1) imagem4;
(2,1) imagem5;
(1,2) imagem6;
(2,2) imagem1;

end tessellation;
```

As Fotos 5.11 e 5.12 mostram instâncias da composição de um mesmo padrão obtidas a partir da primitiva *t.lista*.

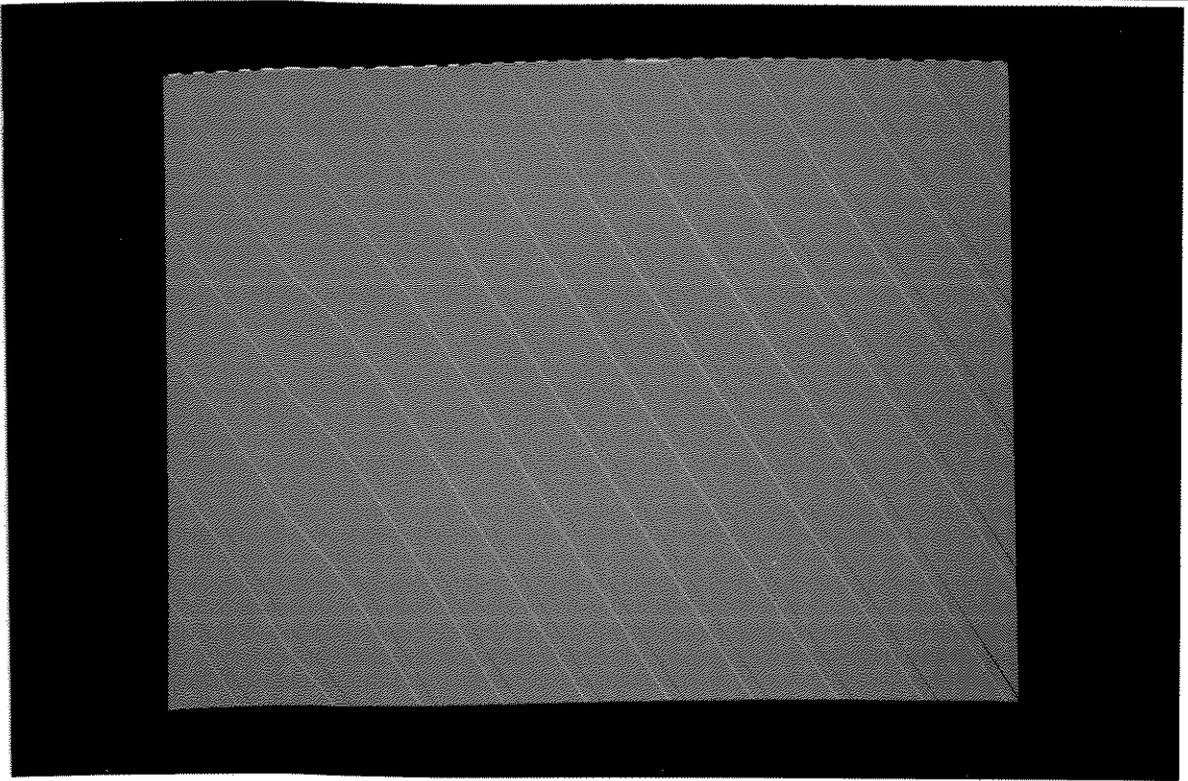


Figura 5.11:Repetições de uma primitiva *t.lista*.

5.3 Fase de Aplicação da Textura

5.3.1 Correspondência Espacial por Mapeamento

O mapeamento é a técnica que permite “colocar” a textura na superfície dos objetos. Considera-se, no algoritmo de mapeamento implementado três espaços: O espaço da imagem (*screen-space*), com variáveis (x, y) ; o espaço de definição do objeto (*object-space*), com variáveis (X, Y, Z) e o espaço de definição da textura (*espaço da textura*), com variáveis (u, v) , geralmente com domínio no espaço unitário $[0, 1]$.

Neste trabalho adotou-se a técnica de mapeamento inverso, em que somente são texturizados os pontos visíveis e, portanto, começa-se no espaço da tela, onde os *pixels* visíveis são levados ao espaço do objeto e então, inversamente mapeados ao espaço da textura.

Aplicar uma textura via técnica de mapeamento consiste em estabelecer a correspondência de pontos entre os espaços objeto-textura, então o problema de texturização

5.3 Fase de Aplicação da Textura

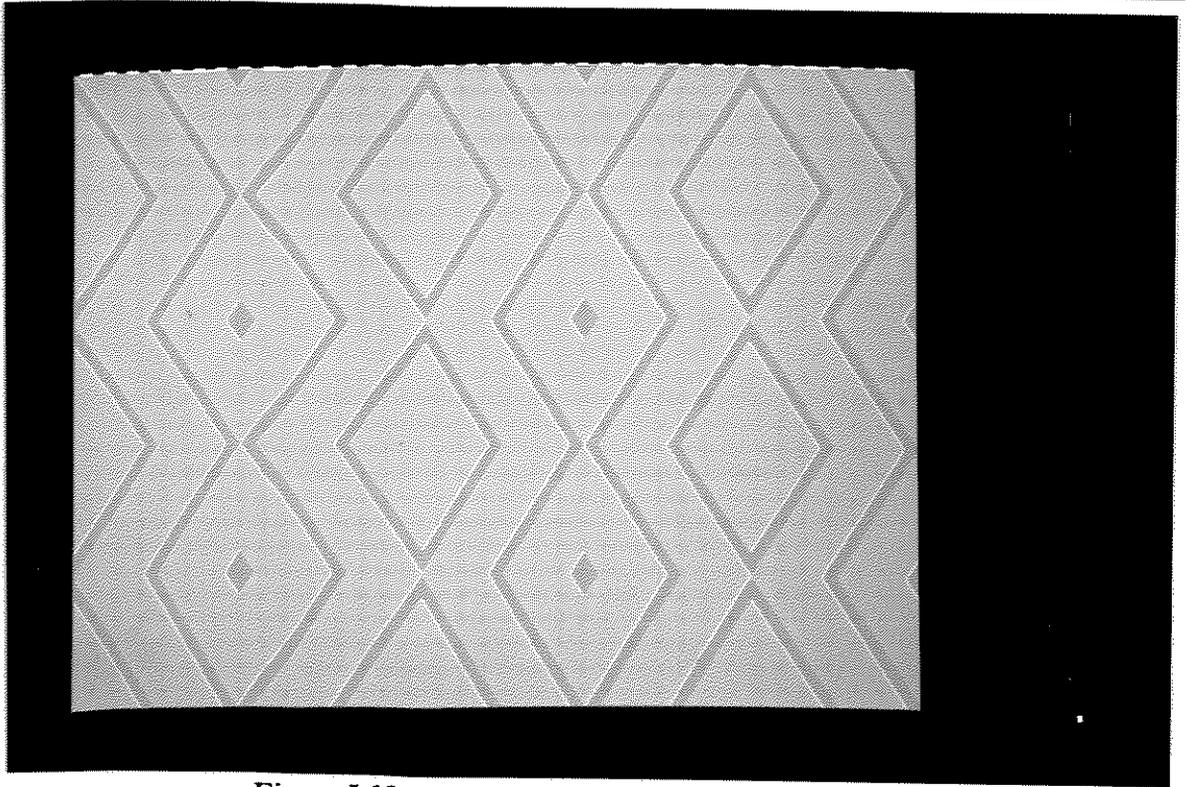


Figura 5.12:A mesma primitiva *t.lista* repetida em espelho.

se reduz a transformar um sistema de coordenadas em outro. Ainda que isto não seja uma regra, geralmente, por simplicidade, se tem assumido uma função de mapeamento linear. Assim, por exemplo, sejam as variáveis de definição do objeto (θ, ϕ) e as de textura (u, v) , o mapeamento consiste em estabelecer as relações:

$$\theta = f(u, v) \quad \text{e} \quad \phi = g(u, v)$$

$$u = r(\theta, \phi) \quad \text{e} \quad v = s(\theta, \phi)$$

onde f e g ou r e s , ao considerar-se lineares, tomam a forma

$$\theta = Au + B \quad \text{e} \quad \phi = Cv + D$$

Nem sempre as coordenadas de definição do objeto X, Y e Z simplificam as relações de mapeamento e dependendo das primitivas, em alguns casos é preferível usar coordenadas paramétricas da primitiva, que as cartesianas (por exemplo, coordenadas esféricas quando a primitiva é uma esfera).

A correspondência entre pontos dos espaços textura-objeto é particular para cada primitiva e, portanto, nos módulos de funções das primitivas, se tem incluído a sua função de mapeamento respectiva, a saber: *cone_map()*, *cylinder_map()*, *inf_cone_map()*, *sphere_map()*.

TX_mapping Esta função realiza o mapeamento. São considerados dois tipos de mapeamento: o que altera as cores da superfície (o resultado é parecido a colar um desenho elástico sobre a superfície) e o mapeamento que altera a normal.

No primeiro caso, o desenho que se mapeia sobre a superfície pode ser obtido a partir de:

- arquivos de imagens do programa de *rendering*;
- imagens rasterizadas, captadas por *scanner* ou câmera de vídeo;
- arquivos de desenho computadorizado;
- arquivos dos padrões gerados no módulo de prototipação;
- procedimentos que criam texturas no espaço normalizado.

No caso de mapeamento por alteração da normal, este pode ser gerado segundo:

- uma função matemática;
- procedimento de alteração da normal;
- algum arquivo de imagem e um procedimento associado (por exemplo converter variações nas mudanças de intensidade nos pixels da imagem em desvios de normal);
- uma tabela de valores de direção da normal.

Parâmetros da Função: Desde o módulo de textura do *raytracing* e de acordo com o identificador da primitiva, é chamada a função de mapeamento de cada primitiva, que recebe como parâmetro as estruturas *Map_Parms* que contém:

n = identificador de face da primitiva;

5.3 Fase de Aplicação da Textura

u, v = variáveis do espaço da textura;

P = estrutura tipo *point* de valores x, y e z , que dão as coordenadas, no espaço objeto, do ponto a ser inversamente mapeado;

N = normal ao ponto P ;

N_{local} = Normal da face;

surf = estrutura do tipo *surface* que contém especificações da superfície à que pertence o ponto P (K_a, K_d, K_s, n, K_t , etc).

Por estar esta função diretamente ligada à aplicação das texturas durante o *rendering*, vários dos seus parâmetros são obtidos durante este processo.

A função devolve valores u, v , do espaço unitário, os quais são usados para acessar as primitivas geradoras de textura, sejam estas compostas ou simples.

Uma descrição dos possíveis parâmetros que podem ser colocados como uma extensão à função de mapeamento é especificada a seguir:

- Ponto no espaço objeto $P(x, y, z)$;
- Tipo de primitiva;
- Face da primitiva;
- Modelo de iluminação;
- Parâmetros do modelo que serão afetados pelo mapeamento;
- Nome do arquivo ou função que contém a textura a ser mapeada;

Caso seja um arquivo, este deve ter um formato padrão; nesta implementação seriam na realidade três arquivos R,G e B.

O comprimento da entrada por pixel é de um byte e os dados são gravados em formato binário. Em caso de ser função, podem ser especificados parâmetros de chamada, caso se deseje um valor diferente ao *default*.

- Área do espaço da textura a ser mapeada;

A janela de texturização é descrita pelos pontos $P_1(u_1, v_1)$ e $P_2(u_2, v_2)$, que correspondem aos vértices superior esquerdo e inferior direito respectivamente.

- *Flag* para mapeamento elástico,

Se o arquivo da textura é uma composição de padrões, este *flag* indica que se deseja que o tamanho do padrão permaneça fixo, independentemente da área sobre a qual vai ser mapeada a imagem (no caso não elástico o padrão de tamanho fixo deverá se repetir um número variável de vezes, segundo a extensão da área);

- Tipo de relação entre as coordenadas (u, v) do espaço da textura e as coordenadas paramétricas do objeto;

Default: LINEAR

- Tipo de projeção da textura sobre o objeto;

Default: ORTOGONAL

5.3.2 Correspondência Espacial por Textura Sólida

O algoritmo de aplicação de textura sólida consiste numa extensão da função de mapeamento, porém é necessário que o espaço de textura tenha sido definido em três dimensões e que cada ponto do espaço $P(x, y, z)$ seja relacionado às coordenadas (u, v, w) do espaço da textura. Este relacionamento pode ser direto, caso a função de textura esteja definida em todo o espaço objeto, ou seja: $u = x, v = y, w = z$; ou através de uma função de escalamento adequado, quando a textura está contida no espaço unitário 3D. Isto é possível considerando-se as coordenadas:

$$(X_{min}, Y_{min}, Z_{min}) \text{ e } (X_{max}, Y_{max}, Z_{max})$$

do volume que envolve o objeto (*bounding box*). Assim, as coordenadas a serem usadas para acessar o espaço das texturas são:

$$u = \frac{X - X_{min}}{X_{max} - X_{min}}, \quad v = \frac{Y - Y_{min}}{Y_{max} - Y_{min}}, \quad w = \frac{Z - Z_{min}}{Z_{max} - Z_{min}}$$

Quando as coordenadas do objeto correspondem ao espaço unitário de definição do objeto nenhuma conversão é necessária.

A diferença do mapeamento 2D, nesta abordagem, não é necessário determinar coordenadas paramétricas, que relacione o espaço tridimensional do objeto ao da textura

e que, portanto, a texturização do objeto é feita de maneira independente à “forma” do objeto.

5.3.3 Funções de Aplicação

A aplicação de textura aos objetos é feita através das funções de aplicação $g_i(f_i(p_i))$, melhor explicadas em 4.2 e definidas pelo usuário através do arquivo editor de aplicação de texturas.

Pela execução de algum algoritmo de geração de texturas (o que chamaremos “primitivas de textura”) são criados os campos texturais. Durante o *rendering*, o valor da textura escolhida para o ponto $P(x, y, z)$ é acessado através do parâmetros (u, v, w) , obtidos como resultados da função de mapeamento. Na verdade, tal como os algoritmos de textura foram implementados, o valor da textura devolvido $T(u, v, w)$ pode ser ou um valor real ou um número inteiro que identifica uma subregião da textura e é apenas através das funções de aplicação de textura g_i que se definem:

- qualitativamente os parâmetros do modelo de iluminação afetados pela textura;
- quantitativamente a forma em que os parâmetros variarão e em que medida.

Quando todas as funções de aplicação foram definidas para todos os parâmetros de tonalização, então o conjunto de todas elas constitui a textura final para o objeto.

Portanto, a função g_i , da textura T , está implementada como uma matriz tridimensional $g[\alpha][\beta][\gamma]$, onde:

g é o nome da textura final, no editor de textura;

α é o identificador da subregião textural;

β é o identificador do parâmetro de tonalização a ser influenciado pela textura (definição qualitativa do modelo de iluminação);

γ é a função $h(\beta)$, que descreve o tratamento a ser dado ao parâmetro identificado em β (definição quantitativa de variação do parâmetro).

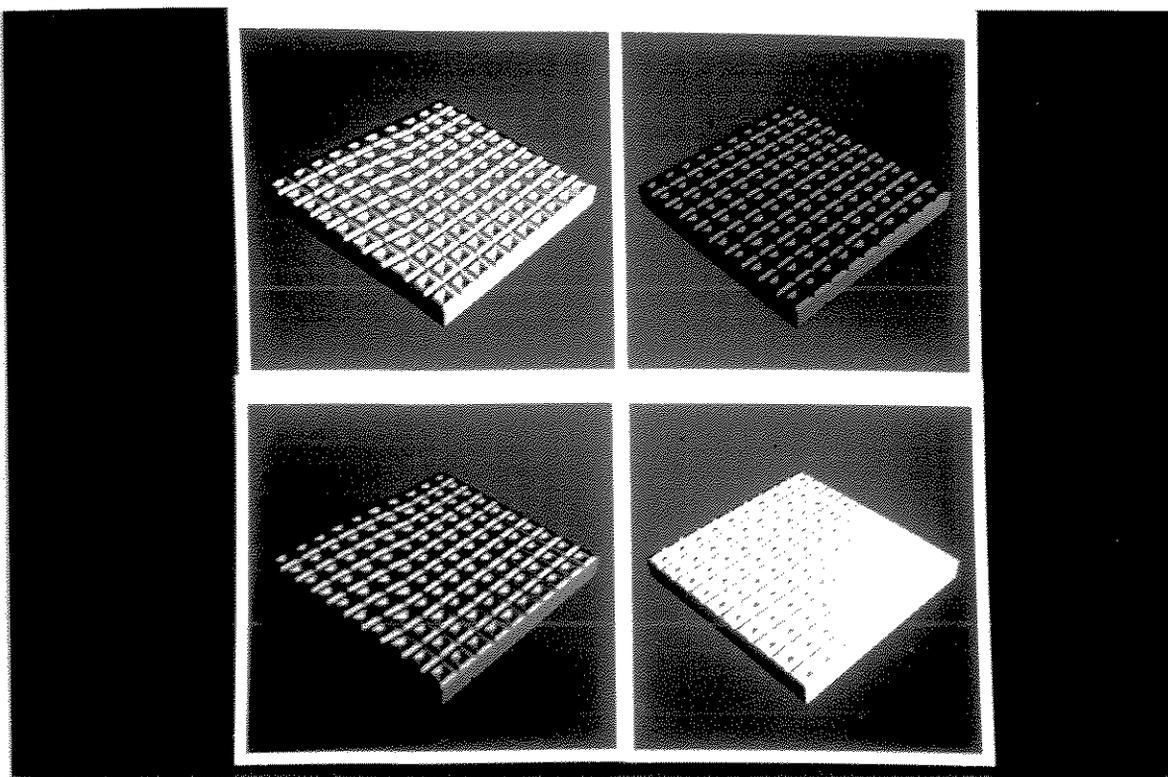


Figura 5.13: Efeitos de variar a função de aplicação.

É interessante notar que a função $h(\beta)$ pode ser um valor constante para o parâmetro β (por exemplo, $\beta = 0.34$), uma operação aritmética sobre o valor de β antes da texturização⁴ (por exemplo, $\beta = 0.1\beta_{inicial} + 0.5$) ou pode conter o identificador de uma outra função de textura (caso de cascata de texturas).

Caso h seja o identificador de uma outra textura T_2 , deverá ser especificada uma outra função de aplicação g_j , com a condição de que, para g_j tem-se $\beta = NULL$, pois o parâmetro já foi definido pela inicial g_i . Isto pode ser entendido como a implementação de um efeito cascata ou *pipeline* de texturas.

A foto 5.13 mostra os efeitos de variar a função de aplicação de textura sobre um objeto, dado um mapa de textura básico.

⁴o valor de β antes da texturização ($\beta_{inicial}$) é o valor do parâmetro para pontos não texturizados do objeto.

5.4 Observações e Comentários

Existem muitas possibilidades para a geração de texturas e como exemplo disto, foram implementados modelos determinísticos e estocásticos.

Duas foram as motivações na implementação descrita:

1. Fornecer primitivas versáteis;
2. Desenvolver e pesquisar ferramentas que multipliquem a capacidade de modelar campos de textura.

A chamada às primitivas com parâmetros de entrada possibilitou satisfazer a primeira das motivações. Como ferramentas úteis: a composição de padrões de textura, a composição de primitivas e finalmente o uso de filtros e técnicas próprias de processamento de imagens mostraram-se muito necessárias à modelagem do campo textural.

Algumas das funções possuem muitos elementos que ao considerá-los variáveis provocam efeitos importantes no resultado obtido. É muito interessante provar e analisar o comportamento os elementos variáveis destas funções a fim de prever os os efeitos na imagem final para conseguir orientar melhor o usuário que deseja modelar uma certa textura. Esta orientação pode estar logo contida em janelas de *help* da interface de comunicação com o usuário.

A experimentação e combinação com as texturas implementadas permite grandes possibilidades de variação.

Capítulo 6

Conclusões

6.1 Conclusões Gerais

Este trabalho abordou o problema da textura em síntese de imagens tomando por base quatro aspectos principais:

1. O estudo de como a textura tem sido tradicionalmente tratada apartir da literatura pertinente.
2. A conceitualização da textura observada através de suas características ópticas, produto da interação da energia luminosa com os objetos do mundo material.
3. Estudo e implementação das técnicas de modelagem do espaço textural bem como dos algoritmos que relacionam o espaço objeto ao espaço de definição da textura.
4. Um estudo e definição de conceitos que permitam ver a textura na sua generalidade. Identificação das fases necessárias à sua síntese bem como uma organização adequada à definição do módulo de texturas num ambiente de síntese de imagens de forma a obter uma solução abrangente ao problema de textura.

Como resultado desta abordagem ficou evidente que:

- A síntese de texturas deve ser resolvida desde dois aspectos: a modelagem do campo da textura e aplicação da mesma aos objetos sintéticos (isto envolve o relacionamento entre o espaço de definição da cena e o espaço de definição da textura).

- Nem sempre é possível tratar do fenômeno textura como algo restrito à superfície do objeto e texturas de diferentes naturezas precisam métodos e técnicas particulares para sua síntese.
- Que os modelos de iluminação em uso nos ambientes de síntese de imagens são simplificações do comportamento real dos raios luminosos e que isto limita a representação de efeitos por exemplo os devidos à natureza ondulatória da luz.
- Também estes modelos não consideram todas as variáveis ligadas ao fenômeno de iluminação que poderiam ser usadas numa representação minuciosa da textura.
- Os materiais são modelados pelo comportamento que eles provocam no raio luminoso e não tanto pela estrutura interna dos mesmos.
- A textura observada pode efetivamente ser conceituada como aquilo que percebemos como resultado da interação entre os raios luminoso e objetos materiais e que é natural ao ser humano reconhecer características ópticas na textura.
- Que é possível descrever uma textura através da descrição das características ópticas reconhecidas nela.
- É possível reconhecer mais de uma escala na textura observada e que é desejável estabelecer os valores dos parâmetros úteis para a síntese de cada escala.
- Dispor de primitivas versáteis promove a criação de grande variedade de texturas o qual é desejável em síntese de imagens.
- Que nem sempre as primitivas de modelagem conseguem representar o efeito desejado e que para isto é útil ter ferramentas para modificar interativamente o mapa de textura até conseguir uma amostra razoável da textura alvo.
- Que para relacionar mapas de textura ao objeto é necessário um procedimento intermediário que defina qualitativa e quantitativamente como o mapa afeta ao objeto e isto é chamado de função de aplicação da textura.

A proposta apresentada baseou-se nos pontos abaixo destacados:

- Que sejam bem distinguidas durante a síntese da textura a fase de modelagem da fase de aplicação da textura e que neste sentido a modelagem do mapa de

textura possa ser feita num ambiente independente ao *rendering*. As variáveis relativas ao modelo de iluminação somente serão consideradas na definição das funções de aplicação do mapa de textura.

Como na percepção da textura é possível identificar a variação espacial das características ópticas presentes na textura e é possível também descrevê-la ao *rendering* através destas características, por tanto, para sintetizar textura:

- * Deve-se estabelecer quais variáveis do modelo de iluminação são úteis para a simulação de cada característica óptica.
 - * Deve-se definir o relacionamento entre estas variáveis de forma a manter coerência com o modelo de iluminação que as usa.
 - * A modelagem da textura deve consistir em criar mapas de textura de cada característica óptica para logo associar estes mapas ao objeto da cena através de funções de aplicação. Assim quando mais de uma característica óptica se faça presente num ponto, funções de mapeamento apropriadas devem acessar o ponto correspondente no espaço da textura e ser relacionado o valor nele contido aos parâmetros de iluminação através das funções de aplicação. A textura final do objeto somente é definida na especificação da função completa de textura (conjunto de funções de aplicação para todos os mapas considerados).
- Que a criação de mapas de textura seja feita através de primitivas flexíveis de geração do campo textural e de ferramentas que transformem o campo textural até alcançar a representação desejada. Também que mapas e primitivas possam ser compostos ou combinados na busca de versatilidade para a modelagem.
 - Considerar a criação de funções intermediárias chamadas de funções de aplicação da textura que permitam a geração de mais de uma textura a partir de um mesmo mapa.
 - Conceituar a texturização como um processo executado durante o *rendering*, que consiste na alteração dos parâmetros do modelo de iluminação no ponto em tratamento através de variações de cor, transparência, brilho, rugosidade ou refletividade, obedecendo ao comportamento definido pela função de aplicação de textura e os mapas de textura.

- Que os parâmetros do modelo de iluminação que sejam susceptíveis de variação devido à texturização, estejam contidos numa estrutura de dados acessível ao módulo de aplicação de textura. Se mais de um modelo de iluminação estiver disponível o módulo deve prever interfaces de comunicação apropriadas a cada um.
- Que se procure a definição de interfaces mais amigáveis ao usuário para a definição das funções de aplicação como bancos de materiais ou bancos de tipos de superfície.

Os principais resultados obtidos com este trabalho são resumidos a seguir:

- Foram definidas as fases necessárias à síntese de textura (modelagem e aplicação da textura) e uma taxonomia dos trabalhos desenvolvidos na área, como uma forma de organizar a informação encontrada.
- O estudo da textura como um fenômeno físico que envolve inúmeras variáveis nos permitiu perceber:
 - * As necessidades em termos do modelo de iluminação do modelo de material e do modelo de percepção visual para um modelamento minucioso da textura;
 - * o porque efeitos de difração, polarização e interferência não conseguem ser modelados com os modelos de iluminação convencionais;
 - * o procedimento necessário para a representação de efeitos como a isotropia presente em algumas texturas e o efeito de resolução da textura;
 - * alternativas para melhorar a conceitualização da textura no *renderer*: um modelo mais apropriado do comportamento dos raios luminosos, um modelo de material mais completo, um estudo mais detalhado da BRDF para texturas mais complexas etc.
- Foram definidos os conceitos de: textura na linguagem natural, textura observada, característica óptica, resolução da textura e domínio da textura observada.
- Foi obtida experiência na modelagem e aplicação de algumas texturas.

- Foram implementadas primitivas flexíveis acessáveis desde a linha de comando para a modelagem de texturas determinísticas e estocásticas assim como um módulo para a composição de padrões de texturas implementando-se neste algoritmos de rotação, escalamento e manipulação de bit-map de imagens.
- Foram implementadas funções de aplicação da textura dentro do *renderer* no módulo *rt_text* da biblioteca do ProSim.
- Foi implementado o relacionamento do espaço objeto ao espaço da textura através do mapeamento tri-dimensional (textura sólida).
- Foram realizadas experiências para a obtenção de amostras de textura a partir de sua descrição no espaço das frequências.
- Foram aplicadas ferramentas próprias ao processamento de imagens (máscaras de detecção de contornos, suavizado de bordas e ruído gaussiano) e evidenciada a utilidade das mesmas durante a modelagem dos mapas de textura.
- Foram realizadas experiências da modelagem de texturas pela manipulação de um espaço de cor escolhido cuidadosamente para a textura que se deseja representar, aplicando sobre estes as primitivas de perturbação até obter o mapa de cor desejado. Desta forma foram obtidas as texturas de mármore e tábua de madeira.
- Foram obtidos resultados iniciais em relação à representação de texturas que imitem "tecidos" (Ver foto 6.1).
- Através do uso de funções de aplicação da textura foram conseguidos "ajustes" no brilho, transparência, rugosidade e cor das superfícies.
- Uma generalização dos conceitos relacionados à textura permitiu conceitualizar um modelo geral, no qual grande variedade de texturas pode ser gerada a partir de algoritmos simples para ser posteriormente adicionada aos objetos da cena como uma característica exclusiva da superfície do objeto (ex. mapeamento tradicional), como uma característica do material do qual é construído o objeto (existe também nas estruturas internas, ex. textura sólida), ou como uma característica possível de ser estendida ao espaço n -dimensional.
- Foi possível simular as características das texturas em síntese de imagens, apoiados na fisiologia de nosso sistema de visão : cada característica óptica pode ser

separadamente representada por modelos de cor, brilho, etc. e logo ser integrado através das **funções completas de textura**, para gerar a impressão visual que caracteriza a textura representada. A impressão final da textura será dada pela abstração do conjunto de características texturais.

- Foi definida uma especificação geral para o Módulo de Texturas no ProSim.
- Algumas imagens sintéticas foram apresentadas como exemplo dos resultados obtidos com esta proposta.

Em nosso entender a proposta apresenta os seguintes pontos positivos ao tratar a questão da textura:

- Fica bem definido "o quê" é textura no escopo deste trabalho, como ela pode ser definida e as fases necessárias para a síntese de texturas de objetos modelados por computador.
- É conseguido diversificar bastante as possibilidades de geração de texturas diferentes a partir de umas poucas primitivas de textura.
- Modelar características ópticas através de funções e logo integrá-las numa textura só é um problema mais fácil de ser abordado, em termos de encontrar funções representativas f_i , do que procurar modelar a textura final diretamente.
- Com o conceito de funções de aplicação, uma mesma primitiva de textura usada de forma diferente dentro do modelo de iluminação pode gerar impressões diferentes. Na verdade, este conceito permitiu-nos abstrair cada representação gráfica no espaço textural como imagens nas quais a textura é vista somente em níveis da intensidade e o que interessa para projetar a impressão final da textura (cor, transparência, brilho, rugosidade) é a diferença entre valores das intensidades em cada posição do espaço textural. A definição específica da impressão resultante sobre as variáveis que definem a textura é conseguida somente na aplicação das funções $g_i(f_i)$.

Uma vantagem de considerarmos estas funções de forma separada é que como a representação do objeto que melhor se assemelhe à realidade é um processo de tentativa e erro, o usuário terá em mãos mais um recurso na procura de

realismo, introduzindo de maneira rápida e simples variações não fundamentais que preservem um padrão base.

- O modelo de textura apresentado serve para definir e sintetizar texturas. Nesta fase do trabalho obteve-se uma grande versatilidade quanto a implementação de qualquer técnica usada na modelagem do mapa de textura, pois o modelo de texturas proposto não provoca dependência do modelo de iluminação usado, facilita sua extensão N -dimensional e satisfaz o objetivo de prover facilidade em multiplicar o número de texturas disponíveis para a síntese, através das funções de aplicação da textura e da composição de padrões ou funções de textura.
- As texturas podem ser geradas a mais de um nível de resolução somente através da composição de primitivas de textura.
- Para adaptar o módulo de síntese de texturas a objetos modelados por técnicas diferentes à considerada (CSG) bastará definir as funções de mapeamento bi-dimesional apropriadas à nova estrutura de dados. A função de mapeamento tri-dimensional é apropriada a qualquer técnica de modelagem.

Sobre os pontos aqui tratados vale a pena ressaltar que:

- A falta de realismo de algumas texturas deve levar a uma avaliação da técnica de modelagem do objeto e da idoneidade do modelo de iluminação em uso.
- Os custos computacionais dos métodos escolhidos para implementar texturas não foram considerados a rigor neste estágio do trabalho. Simplesmente interessou-nos obter experiência na implementação de primitivas segundo o modelo proposto.

6.2 Trabalhos Futuros

Consideramos que existem trabalhos possíveis de serem desenvolvidos como parte exclusiva do módulo de textura e outros cuja implementação completaria as necessidades do mesmo:

- **Trabalhos Futuros no Módulo de Texturas:**
 - * Um campo de experimentação interessante é analisar os resultados de variar outros parâmetros menos convencionais do modelo de iluminação ou mesmo decompor os parâmetros convencionais em sub-partes e analisar os resultados das mudanças por alteração destas. Os resultados podem sugerir novas alternativas no processo de texturização.
 - * A implementação de texturas com mais de um nível de resolução.
 - * A implementação do efeito isotrópico a partir de funções de aplicação da textura.
 - * Definir com base na experimentação, diretivas para a modelagem do espaço textural a partir do espaço das frequências.
 - * Desenvolver simulação de treliças no espaço objeto a partir do mapeamento de texturas.
 - * Desenvolver a modelagem de texturas que simulam tecidos.
 - * Desenvolver primitivas para a manipulação da direcionalidade no espaço de texturas.
 - * Desenvolver texturas dinâmicas no espaço ou no tempo (texturas na animação dos objetos ou o efeito de envelhecimento das texturas).
 - * Algoritmos que relacionem o espaço objeto ao espaço de textura bi-dimensional para a texturização de superfícies definidas por *patches*.
- **Necessidades para o Módulo de Texturas:**
 - * Desenvolver modelos de materiais.
 - * Definir uma base de dados de materiais e tipos de materiais.
 - * Definir com base no uso dos vários modelos de iluminação, os valores e relações entre parâmetros que melhor modelam as características ópticas e efeitos particulares das mesmas.
 - * No caso de texturas mais complexas é necessário o estudo microscópico da estrutura interna de certos materiais, para definir a interação particular da superfície com a energia luminosa (definição da BRDF-*Bidirectional Reflectance Distribution Function*).

- * Cor e estrutura do material atualmente são efeitos que em síntese de imagens se associam durante a modelagem. Na verdade, no fenômeno físico eles estão intimamente ligados (a cor é o produto da interação da luz com o material). Uma modelagem espectral dos raios luminosos bem como uma modelagem orientada a caracterizar os materiais evidenciaria as características ópticas automaticamente.

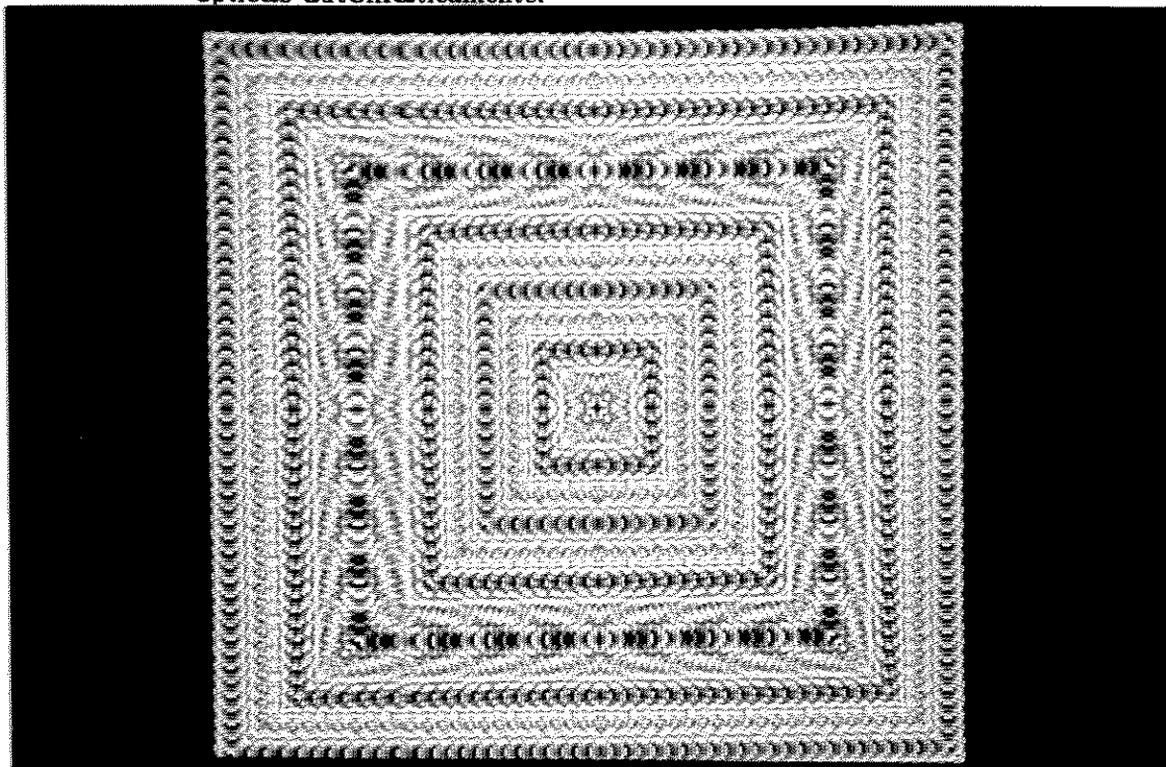


Foto 6.1: Imitação da textura de tecido a partir de uma primitiva fractal

Apêndice A

Algoritmos Geométricos para Geração de Textura:

A seguir estão descritos alguns dos algoritmos implementados para a geração de campos de definição de textura (chamados também "primitivas de texturas").

A.1 Parâmetros Gerais:

u, v são as variáveis para acessar o espaço da textura, sempre com valores entre $[0,1]$ (por convenção);

nu, nv número de repetições do padrão, nos eixos u e v , quando acontece textura por composição do padrão;

default para indicar se, em vez de subregiões de textura ($default = 0$), a função devolverá os valores *default* para colorir as subregiões ($default = 1$);

palette_id para algumas texturas. Ponteiro ao palette de cores;

bit-map a função pode ser chamada via teclado para produzir o *bit-map* da textura desejada ($bit-map = 1$) ou no arquivo editor para ser executada durante o *rendering*.

A.2 Flags Gerais:

Criados através da diretiva de pre-processamento:

#define IDENTIFICADOR (value). Nas funções implementadas estão definidos os seguintes identificadores:

center(*h,k*) Para indicar qual é o centro das coordenadas do padrão textural. Em geral usam-se os pontos (0,0), para parâmetros gerados a partir de coordenadas polares e (0.5,0.5), no espaço unitário da textura;

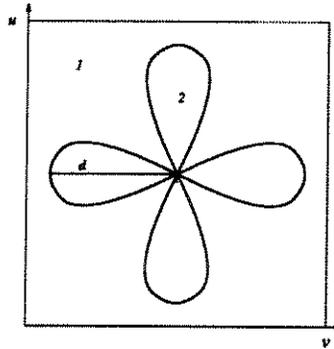
u_dimension, v_dimension Quando o padrão textural não está descrito no espaço unitário da textura, ou seja

$$\text{domínio } u = [0, u_dimension - 1],$$

$$\text{domínio } v = [0, v_dimension - 1].$$

A.3 Primitivas de Textura e Mapas das Subregiões

1) t_flor()



Esta função é implementada a partir da função

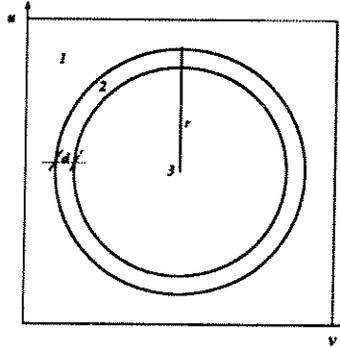
$$r = a \sin(n\theta)$$

e usa os seguintes parâmetros:

n = número de pétalas da flor;

d = longitude das pétalas.

2) `t_ring()`

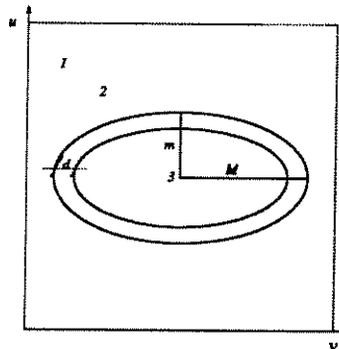


Esta função desenha dois círculos concêntricos separados por uma distância d , definindo três subregiões. Os parâmetros da função são:

r = raio do círculo exterior;

d = espessura do anel.

3) `t_ellipse()`



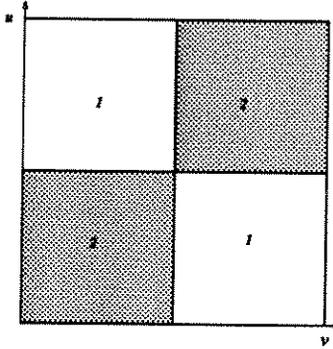
Similar à anterior, porém refere-se a elipses, tendo como parâmetros

M = semi-eixo maior;

m = semi-eixo menor;

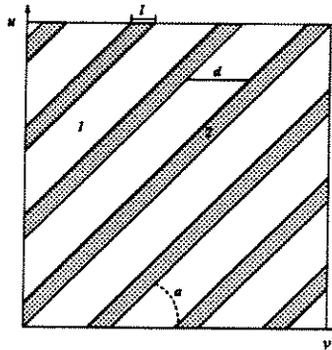
d = espessura do anel.

4) `t_xadrez()`



implementa uma superfície quadriculada, como a de um tabuleiro de xadrez, usando também a equação da reta. São definidos dois tipos de subregiões textural, e não existem parâmetros específicos.

5) `t_listra()`



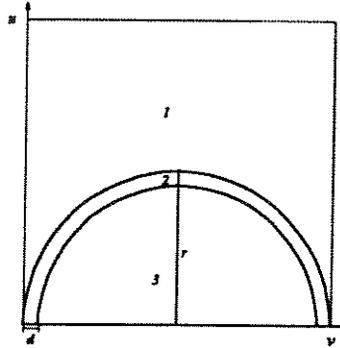
implementa uma listra, delimitada por duas retas paralelas. São definidas duas subregiões. Os parâmetros para esta função são:

a = ângulo com a horizontal, em graus;

l = componente horizontal da largura da listra;

d = componente horizontal da distância entre as listras.

6) **t_ciclóide()**



implementa dois ciclóides e são definidas 3 subregiões. A equação implementada é:

$$x = a \operatorname{acos} \left(\frac{a - y}{a} \pm \sqrt{2ay - y^2} \right)$$

onde $+\sqrt{\quad}$, quando $\theta < \pi \operatorname{rad}$,

$-\sqrt{\quad}$, quando $\theta > \pi \operatorname{rad}$,

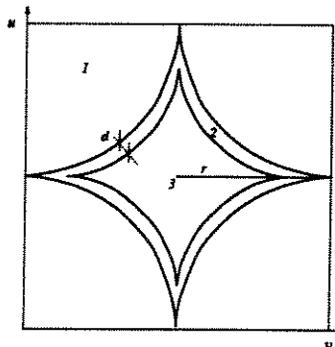
e $a = \frac{aa}{2}$, para $\theta \in [0, 2\pi]$.

Os parâmetros desta função são:

r = raio do ciclóide exterior (em porcentagem);

d = distância ao ciclóide interior.

7) **t_hipociclóide()**



Implementa dois hipociclóides de quatro pontas, definindo três subregiões. Para tal é usado a seguinte equação cartesiana:

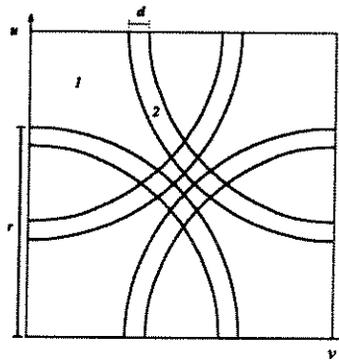
$$x^{\frac{2}{3}} + y^{\frac{2}{3}} = a^{\frac{2}{3}}$$

onde:

r = é o raio desde um vértice ao centro do hipociclóide exterior;

d = a distância entre hipociclóides interior e exterior.

8) `t_four_rings()`



Criado a partir da primitiva `t_ring()`. Gera padrão com duas subregiões, com os parâmetros:

r = raio dos anéis exteriores;

d = espessura dos anéis.

Bibliografia

- [Ama87] John Amanantides. Realism in computer graphics: A survey. *IEEE - Computer Graphics and Applications*, pages 44–58, Janeiro 1987.
- [AS81] Narendra Ahuja and B.J. Schater. Image models. *Computing Surveys*, 13(4):373–393, Dezembro 1981.
- [Bes92] Sergio G. Besuievsky. *Cálculo da Iluminação em Síntese de Imagens através de Ray-Tracing Estocástico - Tese de Mestrado*. Universidade Estadual de Campinas - Inst. de Física, Campinas- São Paulo-Brasil, 1992.
- [BJ83] I.R. Bergen and B. Julesz. Rapid discrimination of visual patterns. *IEEE Transactions on Systems, Man and Cybernetics*, SCM-13(5):857–863, 1983.
- [Bli77] James F. Blinn. Models of light reflection for computer synthesized pictures. *Computer Graphics ACM, SIGGRAPH'77*, pages 192–198, 1977.
- [Bli78] James F. Blinn. Simulation of wrinkled surfaces. Technical report, Caltech, 1978.
- [Bli82] James F. Blinn. Light reflection functions for simulation of clouds and dusty surfaces. *Computer Graphics ACM*, 16(3):21–29, Julho 1982.
- [BN76] James F. Blinn and Martin E. Newel. Texture and reflection in computer generated images. *Communications of the ACM*, 19(10):542–547, Outubro 1976.
- [Bri80] Thomas B. Brill. *LIGHT Its interactions with Art and Antiquities*. Plenum Press, 227 West 17th. Street, New York, N.Y. 10011, 1980.
- [Cat75] Edwin Catmull. Computer display of curved surfaces. *Proceedings of the Conference on Computer Graphics, Pattern Recognition & Data Structure*, pages 11–17, Maio 1975.

- [CG85] Richard J. Carey and Donald P. Greenberg. Textures for realistic image synthesis. *Computer & Graphics*, 9(2):125–138, 1985.
- [Con58] F.W. Constant. *Theoretical Physics: Thermodynamics, Electromagnetism, Waves and Particles*. Addison-Wesley Publishing Company, 1958.
- [Coo84] Robert L. Cook. Shade trees. *Computer Graphics ACM Siggraph'84 Conference Proceedings*, 18(3):223–231, Julho 1984.
- [Cro76] F. C. Crow. The aliasing problem in computers-synthesized shaded image. Technical Report UTEC-CSC-76-015, Cdep. Comptr. Sci., Março 1976.
- [CT81] L. Robert Cook and E. Kenneth Torrance. A reflectance model for computer graphics. *Computer Graphics ACM Siggraph'81 Conference Proceedings*, 15(3):307–316, Agosto 1981.
- [Dur89] Charles X. Durand. Bit-map transformations in computerized 2d animation. *Computer & Graphics*, 13(4):433–440, 1989.
- [dVA80] Coletânea de Vários Autores. *Image Modeling*. Edited by Azriel Rosenfeld - Academy Press, INC, 1980.
- [DvO85] P. Dewaele, L. van Gool, and A. Oosterlinck. Texture analysis anno 1983. *Computer Vision, Graphics and Image Processing*, 29:336–357, 1985.
- [ES89] Gabrielle Englert and Georgios Sakas. A model for description and synthesis of heterogeneous textures. *Proceedings of EUROGRAPHICS'89*, pages 245–256, Setembro 1989.
- [FFC82] Allan Fournier, Don Fusell, and Loren Carpenter. Computer rendering of stochastic models. *Communications of the ACM*, 25(6):371–384, Junho 1982.
- [FL79] K.S. Fu and S.Y. Lu. Computer generation of texture using a syntactic approach. *Computer Graphics and Image Processing*, 9:234–245, 1979.
- [Fra92] Mauricio Jr. França. Projeto prosim - um editor de texturas. Technical report, Relatório Técnico CNPq - LCA - FEE - UNICAMP, Junho 1992.
- [Fu80] K.S. Fu. Syntactic image modeling using stochastic tree grammars. *Computer Graphics and Image Processing*, 12:136–152, 1980.

- [FvFH92] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics principles and practice*. Addison-Wesley, 1992.
- [Gar85] G. Y. Gardner. Visual simulation of clouds. *Computer Graphics*, 19(3):297–303, Julho 1985.
- [Gla88] Andrew S. Glassner. *Course Notes: An Overview of Ray Tracing*. Siggraph'88, 1988.
- [GM85] A. Gagalowicz and Song De Ma. Sequential synthesis of natural textures. *Computer Vision, Graphics and Image Processing*, 30:289–315, 1985.
- [GM86] A. Gagalowicz and Song De Ma. Model driven synthesis of natural textures for 3-d scenes. *Computer & Graphics*, 10(2):161–170, 1986.
- [Gon87] Rafael C. Gonzales. *Digital Image Processing*. Addison Wesley Pub. Co., 1987.
- [GV90] Jonas M. Gomes and Luis C. Velho. *Conceitos Básicos de Computação Gráfica*. VII Escola de Computação Gráfica, São Paulo, 1990.
- [HB84] S. Haruyama and Brian A. Barsky. Using stochastic modeling for texture generation. *IEEE Computer Graphics & Applications*, pages 7–19, Março 1984.
- [HS80] M. Hassner and J. Skalansky. The use of markov random fields as models of texture. *Computer Graphics and Image Processing*, 12:357–370, 1980.
- [Jul75] B. Julesz. Experiments in the visual perception of textures. *Scientific American*, 04:34–43, Abril 1975.
- [KA84] A. Kaufman and S. Azaria. Texture synthesis language for computer generated image. *The fourth Jerusalem Conference on Information Technology*, pages 174–179, Novembro 1984.
- [KA85] A. Kaufman and S. Azaria. Texture synthesis techniques for computer graphics. *Computer & Graphics*, 9(2):139–145, 1985.
- [Kaj85] James T. Kajiya. Anisotropic reflection models. *ACM Siggraph'85 Conference Proceedings*, 19(3):15–22, Julho 1985.
- [Kaj86] James T. Kajiya. The rendering equation. *ACM Siggraph'86 Conference Proceedings*, 20(4):143–150, Agosto 1986.

- [KR78] B.W. Kernighan and D.M. Ritchie. *the C Programming Language*. Prentice-Hall, 1978.
- [Lat87] B. P. Lathi. *an Introduction to Randon Signals and Communication Theory*. International Textbook Company, Electrical Engineering Department - Bradley University, 1987.
- [Lew84] John-Peter Lewis. Texture synthesis for digital painting. *Computer Graphics ACM, SIGGRAPH'84*, 18(3):245-252, Julho 1984.
- [Mad91] Heraldo Madeira. *Referências Verbais*. Unicamp - Laboratório de Computação e Automação - FEE, 1991.
- [Man75] B. B. Mandelbrot. Stochastic models for the earth's relief, the shape and fractal dimension of coastlines and the number area rule for islands. *Proc. Nat. Acad. Sci. USA*, 72(10):2825-2828, 1975.
- [MFV80] J. W. Modestino, R.W. Fries, and A.L. Vickers. Stochastic image models generated by random tessellations in the plane. *Computer Graphics and Image Processing*, 12:74-78, 1980.
- [Min54] M. Minnaert. *LIGHT & COLOR in the open air*. Dover Publications Inc., 180 Varick Street, New York, N.Y. 10014, 1954.
- [MPC74] L. Mezei, M. Puzin, and P. Conroy. Simulation of patterns of nature by computer graphics. *Information Processing*, pages 861-865, 1974.
- [MV68] B. B. Mandelbrot and J. W. Van Ness. Fractional brownian motion, fractional noises and applications. *SIAM Review*, 10(4):422-437, Outubro 1968.
- [Not85] H. Nothdurft. Sensitivity for structure gradient in texture discrimination tasks. *Vision Research*, 25(12):1957-1968, 1985.
- [OS75] A. Oppenheim and R. Schafer. *Digital Signal Processing*. Prentice-Hall, 1975.
- [Pal85] E. Palik. *Handbook of Optical Constants of Solids*. Academic Press, New York, 1985.
- [Pap91] Athanasios Papoulis. *Probability, Random Variables and Stochastic Processes*. Mc Graw-Hill International Editions, 3rd Ed. - 1991.

- [PD84] Thomas Porter and Tom Duff. Compositing digital images. *Computer Graphics - Proceedings of SIGGRAPH 84*, 18(3):253-257, Julho 1984.
- [Pea85] Darwyn R. Peachey. Solid texturing of complex surfaces. *SIGGRAPH'85*, 19(3):279-286, Julho 1985.
- [Per85] Ken Perlin. An image synthesizer. *SIGGRAPH'85*, 19(3):287-291, Julho 1985.
- [Per89] Ken Perlin. Hypertexture. *Computer & Graphics*, 23(3):253-261, Julho 1989.
- [Pho75] Bui-Tuong Phong. Illumination for computer generated images. *Communications of the ACM*, 18(6):311-317, Junho 1975.
- [Ram82] Nevadia Ramakat. *Machine Perception*. Prentice-Hall, Inc, Englewoog Cliffs, New Jersey 07632, 1st Ed. - 1982.
- [Ree83] William T. Reeves. Particle systems: A technique for modeling a class of fuzzy objects. *Computer & Graphics*, 17(3):359-376, Julho 1983.
- [Rog85] D.F Rogers. *Procedural Elements for Computer Graphics*. McGraw Hill, 1985.
- [SA79] B.J. Schater and N. Ahuja. Random pattern generation process. *Computer Graphics and Image Processing*, 10(2):95-114, 1979.
- [SG86] Song De Ma and A. Gagalowicz. Determination of local coordinate systems for texture synthesis on 3-d surfaces. *Computer & Graphics*, 10(2):171-176, 1986.
- [Sim91] Karl Sims. Artificial evolution for computer graphics. *Computer Graphics - Proceedings of SIGGRAPH 91*, 25(4):319-328, Julho 1991.
- [Son90] Graciano Jr. Sonego. Manual de utilização do sistema prosim/raytrace. Technical report, Unicamp - Laboratorio de Computação e Automação - FEE, 1990.
- [SS91] Peter Shirley S. *Physically Based Lighting Calculations for Computer Graphics*. Tese de Doutorado - Universidade de Illinois, 1991.
- [Ste64] Sölve Stenström. *Optics and the eye*. Butterworth & Co. Ltda, London 88 Kingsway W.C. 2, 1964.
- [Tre85] A. Treisman. Preattentive processing in vision. *Computer Vision, Graphics and Image Processing*, 31:156-177, Julho 1985.

- [TS67] K.E. Torrance and E.M. Sparrow. Polarization, directional distribution and off-specular reflection from roughened surfaces. *Journal Optical Soc. Am.*, 57(9):1105-1114, Setembro 1967.
- [UU80] B. Upton and J. Upton. *Photography*. Little Brown and Co., 2nd Ed. - 1980.
- [WAT92] Stephen H. Westin, James R. Arvo, and Kenneth E. Torrance. Predicting reflectance functions from complex surfaces. *Computer Graphics ACM Siggraph'92 Conference Proceedings*, 26(2):255-264, Julho 1992.
- [WB72] Charles S. Williams and Orville A. Becklund. *OPTICS: A short course for Engineers & Scientists*. Wiley-Interscience div of John Wiley & Sons, Inc., 1972.
- [WBC87] C. Wyvill, Wyvill B., and McPheeters C. Solid texturing of soft objects. *IEEE Computer Graphics and Applications*, 7(12):20-26, Dezembro 1987.
- [Whi80] Turner Whitted. An improved illumination model for shaded display. *Communications of the ACM*, 23(6):343-349, Junho 1980.
- [WK91] Andrew Witkin and Michael Kass. Reaction - diffusion textures. *Computer Graphics ACM Siggraph'91 Conference Proceedings*, 25(4):299-308, Julho 1991.
- [Yes79] C.I. Yessios. Computer drafting of stones, wood, plant and ground material. *Computer Graphics*, 13(2):190-198, 1979.
- [YH78] R. Yokohama and R.M. Haralick. Texture synthesis using a growth model. *Computer Graphics and Image Processing*, 8:369-386, 1978.